ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Efficient Algorithms for Incessant Position Based Spatial Queries in Mobile Environments

G.Elavarasi[1], K.Manokaran[2]

Second year M.E. (CSE), Sri Vidya College of Engineering and Technology, Virudhunagar, Tamilnadu, India[1]

Assistant Professor, CSE, Sri Vidya College of Engineering and Technology, Virudhunagar, Tamilnadu, India[2]

ABSTRACT- There is an increase in the process of caching valid regions of location based queries (spatial queries) at mobile clients. It is effective in reducing when the number of queries is posted by mobile clients and the sending queries are loaded on the server. Due to this mobile clients are suffered. That is, mobile clients are waiting for a long time for the server to answer for the posted queries. The proxy-based approach to continuous nearest-neighbor (NN) and window queries was proposed. Proxy creates the Estimated Valid Regions (EVR) for mobile clients by extracting the location and temporal locality of location based queries..In this paper, for window queries to represent the EVR's in the form of vectors called estimated window vectors (EWVs).It is used for estimating the larger valid regions. In distinctly, separate index structures were used namely EVR-tree and Grid index for NN and window queries. To increase the efficiency, we develop algorithms for exploiting the results of NN queries and window queries.

KEYWORDS-Nearest neighbor query, window query, spatial query processing, location-based service, proxy server.

I. INTRODUCTION

Location-based services (LBS) are a general class of computer program-level services used to include specific controls for location and time data as control features in computer programs.LBS is an information service and has a number of uses in social networking today as an entertainment service, which is accessible with mobile devices through the mobile network and which uses information on the geographical position of the mobile device. LBS can provide helpful information regarding public transportation, route options, weather forecasts, and location of hospitals, restaurants, police stations, tourist attractions, landmarks, petrol pumps, ATMs etc. According to spatial constraints, spatial queries can be divided into several categories including nearest neighbor (NN) queries and window queries. An NN query is to discover the closest data object with respect to the location at which the query is issued. For example, a user may launch an NN query like "show the nearest restaurant with respect to my current location." a window query is to find all the objects within a specific window frame. The existing system is based on naïve method. The naive method answering continuous spatial queries is to submit a new query whenever the query location changes. The naïve method is capable to provide correct results, but it has the following limitations:

High power consumption and heavy server load. Another one is, when the number of queries increases, the average waiting time of clients becomes longer. In this, VR computation requires extra I/O and computational cost. It is not suitable for large LBS server.

In this paper, we proposed a proxy-based approach to continuous NN and window queries in mobile environments. The proxy imposes upon spatial and temporal locality of spatial queries to create EVRs of NN and window queries. We introduced new EVR creation and extension algorithms for NN queries, enabling the proxy to build effective EVRs efficiently. The devised algorithms make the proxy achieve high performance even when the cache size is small. We also propose to represent EVRs of window queries in the form of vectors, called estimated window vectors, to achieve larger estimate region.

II. PRELIMINARIES

2.1 Related Work

Location-Dependent Information Services

Location-Dependent Information Services (LDISs) are an important class of context-aware applications. They answer location-related queries, where a location is either explicit or implied. These services emerged from advances and convergence in high speed wireless networks, personal portable devices, and location-identification techniques. In [1],Dik Lun Lee, Jianliang Xu, and Baihua Zheng, proposed location-dependent information access in a mobile-pervasive environment, in particular in a cellular mobile system, and present new research issues arising from on-demand access, broadcast, and data caching. The geometric model's main advantage is its compatibility across heterogeneous systems. The disadvantage is that it is not an sufficient method for finding the spatial queries.

In [2] Baihua Zheng, Jianliang Xu, Student Member, IEEE, and Dik L. Lee studied the issues of cache invalidation and cache replacement for location-dependent data under a geometric location model. They introduce a new performance criterion, called caching efficiency, and propose a generic method for location-dependent cache invalidation strategies. In addition, two cache replacement policies, PA and PAID, are proposed. There are two situations where validity checking is necessary: 1) the same query may be issued later when the client has moved to a new location; 2) a mobile client may keep moving after it submits a query and the client may have moved to a new location when the response is returned if there is a long data access delay. The invalidation information can be utilized by cache replacement policies to enhance performance. The factor of data distance is sensitive to scope distributions, query patterns, and movement models. The disadvantage is that there is no frequent update of data objects.

Range-Monitoring Queries

Range queries have been studied extensively in geographic information systems (GIS), computer aided design (CAD), and other conventional database systems. In these systems, updates are infrequent and the database is relatively stable. In [3], Y. Cai, K.A. Hua, and G. Cao, proposed how to leverage heterogeneous mobile computing capability for efficient processing of real-time range-monitoring queries .The study shows that the new technique is many times better in reducing mobile communication and server processing costs. Provide accurate and real-time query results without requiring the constant location updates from mobile objects .It minimizes the mobile communication overhead. MQM is highly scalable. It requires high input and output computation.

In [4] B. Gedik and L. Liu proposed MobiEyes, Location monitoring is an important issue for real time management of mobile object positions. Significant research efforts have been dedicated to techniques for efficient processing of spatial continuous queries on moving objects in a centralized location monitoring system. Surprisingly, very few have promoted a distributed approach to real-time location monitoring. Most of the existing approaches for processing spatial queries on moving objects are not scalable, due to their inherent assumption that location monitoring and communications of mobile objects are controlled by a central server. The centralized approaches can suffer from dramatic performance degradation in terms of server load and network bandwidth.

In this they presented MobiEyes, a distributed solution for processing MQs in a mobile setup. Our solution ships some part of the query processing down to the moving objects, and the server mainly acts as a mediator between moving objects. This significantly reduces the load on the server side and also results in savings on the communication between moving objects and the server.The distributed processing of MQs significantly decreases the server load and scales well in terms of messaging cost while placing only small amount of processing burden on moving objects. The MobiEyes solution

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

discussed in this paper focuses on real-time evaluation of moving queries in real-world settings, where the trajectories of the moving objects are unpredictable and the queries are associated with moving objects inside the system.

Time-Parameterized Queries

In [5], H. Hu, J. Xu, and D.L. Lee proposed a generic framework for monitoring continuous spatial queries over moving objects. The frame-work distinguishes itself from existing work by being the first to address the location update issue and to provide a common interface for monitoring mixed types of queries. Based on the notion of safe region, the client location update strategy is developed based on the queries being monitored. Thus, it significantly reduces the wireless communication and query reevaluation costs required to maintain the up- to-date query results. We propose algorithms for query evaluation/reevaluation and for safe region computation in this framework. Enhancements are also proposed to take advantage of two practical mobility assumptions: maximum speed and steady movement. This paper demonstrates the feasibility and performance advantages of the framework. The safe regions for kNN queries are computed separately; so the final safe region of the object p.sr may not be optimal.

In [6], Y. Tao and D. Papadias proposed a general framework that covers time-parameterized variations of the most common spatial queries, namely window queries, k-nearest neighbors and spatial joins. Time-parameterized queries (TP queries for short) retrieve

- the actual result at the time that the query is issued,
- the validity period of the result given the current motion of the query and the database objects, and
- the change that causes the expiration of the result.

The proposed techniques significantly extend previous work, both in terms of effectiveness and applicability to far more general problems. The disadvantage is that if the time expires for the particular query the process will be skipped out. In [7], S. Nutanong, R. Zhang, E. Tanin, and L. Kulik presented an incremental safe-region-based technique for answering MkNN queries, called the V*-Diagram.A safe region is a set of points where the query point can move without changing the query answer. Voronoi diagram has low input and output and low computation costs compare with other method but it needs large amount of space. Objects location can be stored only in servers, so there will be congestion occurs.

In [8], L. Kulik and E. Tani presented a set of incremental algorithms that track the rank of the neighbors of a moving query point continuously. Nearest neighbor (NN) queries are a well investigated research topic. With the ubiquitous availability of location information for mobile devices, continuous nearest neighbor(CNN) queries have become a major research focus. An important variant of CNNs are continuous ranking queries that retrieve the rank of k neighbors of a moving object. A number of algorithms have been developed to efficiently find the nearest neighbors for a static query point. One way to derive an algorithm for a CNN query is to use a classical NN query algorithm: for each predetermined time or location update interval ,a new NN query for a moving query point can be executed using one of the classical NN query algorithms. However, a repeated execution of the classical algorithms will not take advantage of the steps and the results of previously run NN queries.

In this paper, they presented a set of incremental algorithms that track the rank of the neighbors of a moving query point continuously. The positions of the neighbors are assumed to be fixed. We show that updates for such continuous rank queries can be performed in linear time for any arbitrary polygonal curve without any prior knowledge of the curve itself. For a moving query point, we introduced algorithms to track the ranks of sites without any prior knowledge about the trajectory of the query point. We defined a fixed rank region that determines the set of all those points for which the ranks of the sites do not change. This reduces the need for any query reprocessing and communication because in a fixed-rank region no updates are required. However, the algorithms did not adapt to cope with dynamic situations in which the sites as well as the query point are in motion.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Location-based Spatial Queries

Location-based spatial queries (LBSQs), which refer to a set of spatial queries that retrieve information based on mobile users' current locations. In the simplest approach, a user establishes a point-to-point communication with the server so that her queries can be answered on demand. In [9], W.-S. Ku, R. Zimmermann, and H. Wang presented a novel query processing technique that, while maintaining high scalability and accuracy, manages to reduce the latency considerably in answering location-based spatial queries. Our approach is based on peer-to-peer sharing, which enables us to process queries without delay at a mobile host by using query results cached in its neighboring mobile peers. The advantage of the broadcast model is that it is a scalable approach. By virtue of its peer-to-peer architecture, the method exhibits great scalability: the higher the mobile peer density, the more queries can be answered by peers. Therefore, the query access latency can be markedly decreased with the increase of clients. Its main limitation lies in its lack of security.

In [10], F.P. Tso, J. Teng, W. Jia, and D. Xuan, Proc presented an empirical study on the performance of mobile High Speed Packet Access (HSPA, a 3.5G cellular standard) networks in Hong Kong via extensive field tests. It means includes all possible information about the urban areas like trains, subways, and city buses. It is for finding the areas in various things like from land to sea and from ground(train) to subways. It concludes about is to find the locations, it has been monitored through the HSPA.HSPA identifies the bandwidth, time spending, throughput etc. HSPA monitors the location through the movement of node(object).The limitation is that communication characteristics in HSPA transitional regions are very complicated so more intelligent handoff algorithms are needed for seamless service provisioning. Heavy network traffics will occur in the wireless environment.

III. PROXY BASED APPROACH

A proxy builds EVRs of NN queries and EWVs of window queries based on NN query history and available data objects, respectively. The proxy maintains an object cache and two index structures: an EVR-tree for NN queries and a grid index for window queries. The two index structures share the data objects in the object cache. A proxy server is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers. A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other resource available from a different server and the proxy server evaluates the request as a way to simplify and control its complexity. Proxy creates the Estimated Valid Regions (EVR) for mobile clients by extracting the location and temporal locality of location based queries.

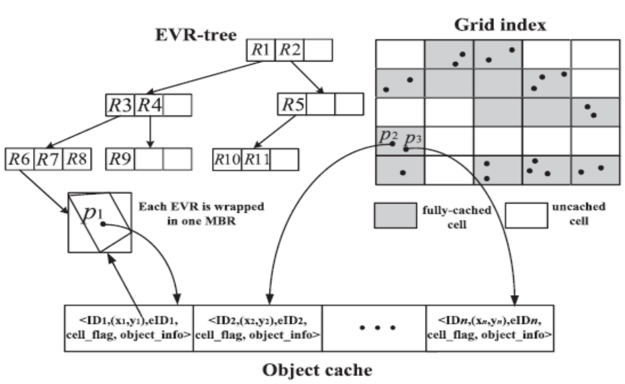

Figure 1. Object cache, EVR-tree, and grid index.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Figure 1 depicts the proposed system architecture for NN and window query processing. The system architecture consists of three parts:
1) an external LBS server, 2) deployed proxies, and 3) the mobile clients.

The LBS server is responsible for managing static data objects and answering the queries submitted by the proxies. Note that the LBS server can use any index structure (e.g., R-tree or grid index) to process spatial queries. The LBS server is assumed not to provide VRs. Each of the deployed proxies supervises one service area and provides EVRs of NN queries and EWVs (vector form of EVRs) of window queries for mobile clients in the service area. Each base station serves as an intermediate relay for queries and query results between mobile clients and the associated proxy. A mobile client maintains a cache to store the query results and the corresponding EVRs. When client has a spatial query, the mobile device first examines whether the current location is in the EVR of the stored result. If so, the stored result remains valid and the mobile device directly shows it to the client. Otherwise, the mobile device submits the query, which is received and then forwarded by the base station, to the proxy. For the received query, the proxy will return the query result as well as the corresponding EVR to the client.

IV. SYSTEM ARCHITECTURE

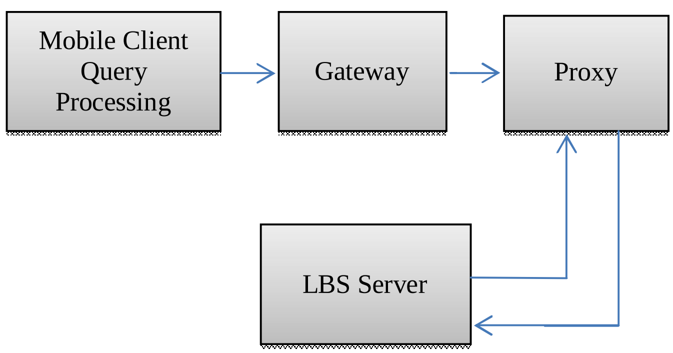

Figure 2 System Architecture.

In Figure 2 the mobile clients' posts query in the mobile client query processing. If the mobile client has a spatial query, it examines the whether it has answer for that query. If it is found, it returns the answer; otherwise it has been posted to gateway based on the Estimated Valid Region cache. It is received by the gateway and it posts query to the proxy. In the query processing, it has been classified into two types. They are NN query and Window query. When the queries are posted, it is received by the gateway. Each gateway acts as an intermediate between query and query results between mobile clients and proxy. Gateway is used for forwarding the queries to the proxy. If proxy returns the result, it has been forwarded to the mobile clients.

A proxy builds EVRs of NN queries. It also builds EWVs of window queries. It is based on NN query history and available data objects, respectively. The proxy maintains an object cache and two index structures. They are EVR-tree for NN queries and a grid index for window queries. In the proxy, it checks whether the query is in any EVR, if it found it return answers to the client .If the query is not found in the EVR, it the proxy attempts to answer the query with the grid index. The proxy extends the received NN query into a 2NN query with the same query location and submits the 2NN query to the LBS server. If it is correct, insert into the cache otherwise reinsertion process takes place.LBS server is nothing but a location based server. It keeps the large amount of location based queries in the static cache. In the LBS server, it has some input parameters such as node id, service, category, region, process.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

We propose two algorithms for NN queries to improve EVR growth. It is also build effective EVR's when the cache size is small. In this paper, for window queries to represent the EVR's in the form of vectors called estimated window vectors(EWVs).It is used for estimating the larger valid regions. In distinctly, we use separate index structures namely EVR-tree and Grid index for NN and window queries. To increase the efficiency, we develop algorithms for exploiting the results of NN queries and window queries. When the number of mobile clients are large, due to the cached EVRs and objects at the proxy, the LBS server of our approach has lower server load. Because of the high client cache hit ratio achieve low server load. Using these algorithms, proposed approach overcome the existing proxy based approaches.

IV. CONCLUSION

In this paper, we have proposed a proxy-based approach to continuous NN and window queries in mobile environments. The proxy imposes spatial and temporal locality of spatial queries to create EVRs of NN and window queries. Furthermore we have examined the impact of data object updates to efficiently handle frequent object updates.

REFERENCES

[1] D. Lee, B. Zheng, and W.-C. Lee, "Data Management in Location-Dependent Information Services," IEEE Pervasive Computing, vol. 1, no. 3, pp. 65-72, July-Sept. 2002.
[2] B. Zheng, J. Xu, and D.L. Lee, "Cache Invalidation and Replacement Strategies for Location-Dependent Data in Mobile Environments," IEEE Trans. Computers, vol. 15, no. 10, pp. 1141-1153, Oct. 2002.
[3] Y. Cai, K.A. Hua, and G. Cao, "Processing Range-Monitoring Queries on Heterogeneous Mobile Objects," Proc. Fifth IEEE Int'l Conf. Mobile Data Management, pp. 27-38, 2004.
[4] B. Gedik and L. Liu, "Mobieyes: A Distributed Location Monitoring Service Using Moving Location Queries," IEEE Trans.Mobile Computing, vol. 5, no. 6, pp. 1384-1042, Oct. 2006.
[5] H. Hu, J. Xu, and D.L. Lee, "A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 479-490, 2005.
[6] Y. Tao and D. Papadias, "Time-Parameterized Queries in Spatio-Temporal Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 334-345, 2002.
[7] S. Nutanong, R. Zhang, E. Tanin, and L. Kulik, "The V _-Diagram:A Query-Dependent Method for Moving kNN Queries," Proc.VLDB Conf., pp. 1095-1106, 2008.
[8] L. Kulik and E. Tanin, "Incremental Rank Updates for Moving Query Points," Proc. Int'l Conf. Geographic, Information Science,pp. 251-268, 2006.
[9] W.-S. Ku, R. Zimmermann, and H. Wang, "Location-Based Spatial Query Processing in Wireless Broadcast Environments," IEEE Trans. Mobile Computing, vol. 7, no. 6, pp. 778-791, June 2008.
[10] F.P. Tso, J. Teng, W. Jia, and D. Xuan, "Mobility: A Double-Edged Sword for Hspa Networks: A Large-Scale Test on Hong Kong Mobile Hspa Networks," Proc. MobiHoc Conf., pp. 81-90, 2010.