ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 2, February 2013

DISTANCE BASED RANGE QUERIES

Josepin Dayana T[1], J. A. M Rexie [2]

1 Post-Graduate Student, Department of Computer Science and Engineering, Karunya University,India
2 Assistant professor, Department of Computer Science and Engineering, Karunya University, India,

Abstract- Nowadays, a Location Based Services (LBS) are gaining more popularity. Spatial data mainly designed to focus on the location-enabled applications and geographic information system. Given a positive value r, a distance-based range query returns the objects that lie within the distance r of the query location. Distance based range queries are continuously change their locations in Euclidean space. In this paper, we continuously monitor the distance based range queries over static and dynamic data objects. Moreover the range query is compared with the kNN query. Our simulation results says that the range query will give the better results compared with kNN query.

Index Terms - Range queries, spatial data, Euclidean space, Spatial Queries.

I. INTRODUCTION

A Spatial Data Base Management System (SDBMS) is software that can work with database management system, supports spatial data model, spatial abstract data types (ADT) and spatial indexing, (e.g.-tree). Traditional non-spatial data base system allows concurrent access of data and only efficient for non-spatial queries. Whereas, spatial database management supports spatial queries (List the names of all bookstores within 10km). It deals with large collections of relatively simple geometric objects. Spatial database contains a set of objects in space rather than images or pictures of a space.

Location services can be defined as services that integrate a mobile device's location or position with other information so as to provide added value to a user. They deliver specific information to their users based on their current location. Examples of such applications include finding the nearest restaurant, delivering weather and/or traffic information, and sending coupons to nearest customers. Location services are mainly used in three areas: military and government industries, emergency services, and the commercial sector.

For the Location Based Services (LSB) we use Global Information System (GIS). The GIS is different from SDBMS. GIS is a software to visualize and analyze spatial data using spatial analysis function (say search, location analysis etc.,) GIS uses SDBMS to store, search, query and share large spatial data's. In modeling of spatial data base system two views are used. They are (i) single objects, and (ii) spatially related collections of objects.

Single objects: The fundamental abstractions are point, line, and region. A point represents (the geometric aspect) an object A line is the basic abstraction for facilities for moving through space, or connections in space (roads, rivers, cables for phone, electricity, etc.). A region is the abstraction for something having an extent in 2d-space, e.g. a country, a lake, or a national park.

Spatial Related Collections of Objects: The two most important instances of spatially related collections of objects are partitions (of the plane) and networks .A partition can be viewed as a set of region objects that are required to be disjoint. Partitions can be used to represent thematic maps. A network can be viewed as a graph.

This paper is organized as follows: Section II includes a discussion on key concepts in this paper; section III gives related works; section IV and V discuss about the continuous kNN query and range query. Section VI and VII gives the system model and experimental results. Section VIII gives the conclusion of this paper.

II. KEY CONCEPTS

Range Queries

A range query is a common database operation that retrieves all records where some value between an upper and lower boundary. In this paper the range queries are used to find the particular location within that range. (E.g. find all hotels within 10kms).

Spatial Data

Spatial data are also known as geospatial data or geographic information. It is the data or information that identifies the geographic location of features and boundaries on earth. There are two types of spatial data. The geometry data type supports planner or

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 2, February 2013

Euclidean (flat-earth) data. The geography data type stores ellipsoidal data (round-earth) such as GPS latitude and longitude coordinates.

Euclidean Space

A Euclidean space is a metric space that's linear and finite dimensional. The space which contains a set of objects or points such that for every pair of points there is a non-negative real number called distance that is symmetric and satisfies the triangle property. In Euclidean space used to find the distance between the pair of objects in the space.

Spatial Queries

Spatial query is a special type of database query supported by geodatabases and spatial database. These queries are differing from SQL queries. Most important are that they allow they use of geometry data types such as points, lines and polygons.

III. RELATED WORKS

There is a large body of research work on spatial temporal query processing. Early work assumed a static dataset and focused on efficient access methods (e.g., R-tree) and query evaluation algorithms. Recently, a lot of attention has been paid to moving-object databases, where data objects or queries (or both) move. Tao et al. [2] introduce Time-Parameterized queries (TP queries). A TP query assumes that the motion pattern (e.g., path and speed) of the query is known and retrieves the current results along with a future time at which the current results will become invalid. Benetis et al. [5] developed query evaluation algorithms for NN and reverse NN search based on the TPR-tree. The first category assumes movement trajectories. Continuous kNN monitoring has been investigated for moving queries over stationary objects and linearly moving objects. Iwerks et al. [7] even extended to monitor distance semi joins for two linearly moving datasets. However, as pointed out in, such an assumption does not hold for many application scenarios (e.g., the velocity may change constantly when a car moves on the road). The second category does not make any assumption on object movement patterns. Xu et al. and Zhang et al. [8] suggested returning to a moving query the current result as well as its validity scope where the result remains the same. The query is reevaluated only when the query exits the validity scope. However, these proposals work for stationary objects only. For continuous monitoring of moving objects, the prevailing approach is periodic reevaluation of queries, at each evaluation step, only those objects that have moved since the previous evaluation step are evaluated against the Q index.

IV. CONTINUOUS MONITORING OF k - NN QUERIES

Consider the problem of continuously monitor the k-NN (k-Nearest Neighbor) queries over moving objects within a two dimensional region. (i.e.) queries are important in many applications. In the gaming example, a player wishes to keep track of the k near-by players in order to make a combat plan. The set of data objects is highly dynamic, each data object can move in an unrestricted fashion. For a query point Q also dynamic. It is necessary to continuously monitoring of exact k Nearest Neighbors (k-NNs) of each query point over time. Since do not make any assumptions on the trajectories of the objects, exact query answering can only be done at the expense of sometime delay. In order to reduce the time delay to continuously monitor the k-NN queries indexing schemes are used [15]. The first scheme is based on indexing the objects themselves called as Object-Indexing and the second scheme is based on indexing the queries called Query-Indexing. In both methods, the index takes the form of a grid structure, which represents a canonical partition of the 2D space. The grid structure is preferable to other types of indices such as R-trees because its simple structure lends itself to fast maintenance, which is a desired property in the presence of highly dynamic data. The grid structure is preferable to other types of indices such as R-trees because its simple structure lends itself to fast maintenance, which is a desired property in the presence of highly dynamic data. In fig1. The index structure consists of each cell (i, j) having an object list, denoted by PL$(i; j)$ containing identifiers (IDs) of objects enclosed by the cell $(i; j)$, namely PL$(i; j)$. The object identifier (ID) is then the position in the array.

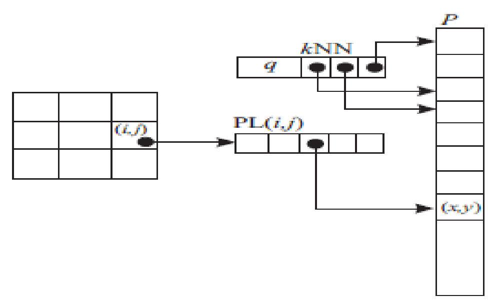

Figure1. Data Structure of the Object Index

V. CONTINUOUS MONITORING OF RANGE QUERIES

With the growing popularity of GPS enabled mobile devices and the advances in wireless technology, the efficient processing of continuous range queries, which is defined as retrieving the information of

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 2, February 2013

moving objects inside a user-defined region and continuously monitoring the change of query results in this region over a certain time period, has been of increasing interest. Continuous range query processing is very important due to its broad application base. The range queries are important type of query, used to find the objects within the range (i.e., "find all the restaurants within 10km"). In case of distance based range queries, consider an object O, a positive value called range r, and a query point q, dist (o,q)≤ r. The dist (o,q)≤ r is defined as the distance between the object O and a query point q, that is less than or equal to range r. Consider the continuous monitoring of moving range queries over static data objects, i.e., a scenario where the queries are constantly moving but the data objects do not change their locations. The following real time example will clearly explain this scenario. Suppose a family travelling by car. They need to reach their final destination by a certain time, while they want to continuously monitor restaurants within 10 km of their current location.

VI. SYSTEM MODEL

In this system model the data objects are indexed with a regular grid, i.e., a partitioning of the data space into square cells of equal size with side-length Ω (a system parameter) [12]. Each cell stores the object coordinates falling inside, and maintains their total number. In figure 2 where the data objects in the system are P_1 to P_n. In this example, cell $C_{0;0}$ contains the coordinates of objects *p*1 and *p*2. The grid information is placed into packets to form the index segment. Each cell may contain any number of objects (or called nodes). Assume in range queries each region is overlapped with the nearby regions.

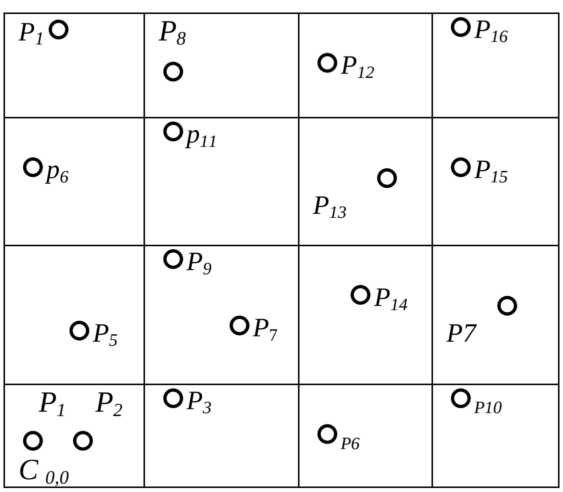

Figure2. Object Index

Query processing. The query can be runs completely at the client side. Let *q* be the client location. Given a cell *c*, *maxdist*(*c*) factor used to find the maximum possible distance between any point in *c* and *q*. Similarly, *mindist*(*c*) is the minimum possible distance between any point in *c* and *q*. If there are at least *k* objects within distance d_{max} from *q*, then a cell *c* (or bucket) does not have to be considered if $mindist(c) \geq d_{max}$, since it cannot contain any better neighbor.

VII. EXPERIMENTAL RESULTS

To implement range queries over static objects we use Microsoft SQL server 2008 as backend and net beans IDE 7.1.2 as front end. The implementation language is JAVA.

In this experiment, take both the data object and query as dynamic (i.e. in our taxi example, cabs may move, new ones may enter service or existing ones may go off duty).The entire space is divided into equally regular grids. Randomly place the nodes inside the each region. The Input of this experiment is the number of objects (or nodes) in each region. The data objects are continuous so they can move inside the query region by setting the time. For every 5 minutes the location of the node is to be changed and also set the transmission range for answering the query to be 50 meters.

The performance can be compared with the kNN query and the range query. The kNN query will sense the nodes within the region and give the answer for the query whereas the range query is to overlap with the nearby regions, hence it will give the result of the query as much as possible and more queries will be answered in range queries. So in case of range queries, we can improve the query answering rate. In kNN query the missing rate is also greater compared with the range queries. So the distance based range queries give the better result to the client.

VIII. CONCLUSION

The distance based range queries are used to answer the query within the specified region. Here both the data objects and query points are dynamic. The distance based range queries are compared with the kNN queries. The range queries are overlapped with the nearby regions. So if the query is given, it will interact with the nearby regions and give the answer for the query, whereas, in kNN query the performance is decreased compared with the range queries. Because in kNN query the query answering rate is decreased, so the client may not get the desired answer. The range queries will give the answer as much as possible.

In future work, the range query may be compared with other different type of queries like NN (Nearest Neighbor) query, RNN (Reverse Nearest Neighbor) query etc.

REFERENCES

[1] Y. Cai, K.A. Hua, and G. Cao, "Processing Range-Monitoring Queries on Heterogeneous Mobile Objects," Proc. Conf. Mobile Data Management, 2004.

[2] Y. Tao and D. Papadias, "Time-parameterized queries in spatio-temporal databases," in *SIGMOD Conference*, 2002, pp. 334–345

[3] H. Hu, J. Xu, and D.L. Lee "A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects," Proc. ACM SIGMOD Conf., pp. 479-490, 2005.

[4] M.A. Cheema, X. Lin, Y. Zhang, W. Wang, and W. Zhang, "Lazy Updates: An Efficient Technique to Continuously Monitoring Reverse knn," Proc Conf. Very Large Database (VLDB), vol. 2, no. 1 pp. 1138-1149, 2009.

[5] S. Nutanong, R. Zhang, E. Tanin, and L. Kulik, "The V*-Diagram: A Query-Dependent Approach to Moving Knn Queries," Proc. Very Large Databases, vol. 1, no. 1, pp. 1095-1106, 2008.

[6] R. Benetis, C. S. Jensen, G. Karciauskas, and S. Saltenis, "Nearest neighbor and reverse nearest neighbor queries for moving objects," in *IDEAS*, 2002, pp. 44–53.

[7] H.-J. Cho and C.-W. Chung, "An Efficient and Scalable Approach to cnn Queries in a Road Network," Proc. Conf. Very Large Databases (VLDB), pp. 865-876, 2005.

[8] F. Liu, T.T. Do, and K.A. Hua, "Dynamic Range Query in Spatial Network Environments," Proc. Database and Expert Systems Applications (DEXA), pp. 254-265, 2006.

[9] G. S. Iwerks, H. Samet, and K. P. Smith,B. "Continuous k-nearest neighbor queries for continuously moving points with updates," in *VLDB*, 2003.

[10] Zheng and D. L. Lee, "Semantic caching in location-dependent query processing," in *SSTD*, 2001, pp. 97–116

[11] M.A. Cheema, L. Brankovic, X. Lin, W. Zhang, and W.W. 0011, "Multi-Guarded Safe Zone: An Effective Technique to Monitor Moving Circular Range Queries," Proc. Int'l Conf. Data Eng. (ICDE), pp. 189-200, 2010.

[12] Continuous Monitoring of Spatial Queries in Wireless Broadcast Environments. Kyriakos Mouratidis, Spiridon Bakiras, Member, IEEE, and Dimitris Papadias--2009.

[13] H.-P. Kriegel, P. Kroger, and M. Renz, "Continuous Proximity Monitoring in Road Networks," Proc. Symp. Geographic Information Systems (GIS), p. 12, 2008.

[14] M. Hasan, M. A. Cheema, X. Lin, and Y. Zhang, "Efficient construction of safe regions for moving knn queries over dynamic datasets," in *SSTD*, 2009, pp. 373–379.

[15] X. Yu, K.Q. Pu, and N. Koudas, "Monitoring k-Nearest Neighbor Queries over Moving Objects," Proc. Int'l Conf. Data Eng. (ICDE), 2005.