IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 5 Issue 4, April 2018
ISSN (Online) 2348 – 7968
www.ijiset.com

Survey on - Facilitating Secure and Efficient Spatial Query Processing on the Cloud

Athulya S.
Computer Science, MES College of Engineering, Kuttipuram
athulyas.devadas@gmail.com

Harikrishnan G. R.
Computer Science, MES College of Engineering, Kuttipuram
harikaipally@gmail.com

Abstract— ***Spatial database systems over the underlying database technology for geo- graphic information systems and other applications. A system must at least be able to retrieve from a large collection of objects in some space those lying within a particular area without scanning the whole set. Therefore spatial indexing is mandatory. It should also support connecting objects from different classes through some spatial relationship in a better way than by altering the relationships that are important for the application. Data owners with large volume of data can outsource spatial databases by taking advantage of the cost effective cloud computing model with effective on-demand features such as scalability and high computing power. Data confidentiality in outsourced databases is a key requirement and therefore untrusted third party service providers in the cloud should not be able to view or manipulate the data. Here the main challenge is maintaining data confidentiality with respect to the cloud service provider as well as providing relevant query results in real-time to authenticated users by space filling curve methods. The approaches till date is either compromising on the data confidentiality or which carries high communication cost between the server and user. Here conducted a study on various methods used in Hilbert curve for achieving data confidentiality along with the less communication cost.***

Keywords-Space filling curve; Hilbert curve

I. INTRODUCTION

In different fields there is need to manage data which are related to space, means data which are in geometric space. The space can be of any parts of the top of the earth that is, geographic space that are readily seen, like a man-made space like the layout of a Very Large Scale Integration (VLSI) design, a volume containing a model of the human brain, or another three dimensional space representing the arrangement of chains of protein molecules[5]. The data which are stored in a spatial database, spatial data, are of mainly two types; point data and region data. Point data are used to specify the values which posses measured value even in pixel values. Where as in region data objects have spatial extent with location and boundary, database typically uses geometric approximations constructed using line segments, polygons, etc.

A cloud architecture holds mainly three entities named Data Owner (DO), Service Provider (SP) Authorized User (AU). DO are meant to manage its data at a third-party cloud Service Provider (SP) responses to the queries of Authorized User (AU). As it is mentioned the SP is a third-party it can't be trusted up to the mark, there are some chances for leakage of query point location of AU. To overcome this issue we are using different conventional encryption schemes such as Advanced Encryption Scheme (AES). Where the data are encrypted; the encryption makes the DO to use or query the same difficult. So at that point it made more crucial to make the data secured at the same time providing location privacy for the confidential data. To make it possible space filling curves have been opted.

A. Space Filling Curves

Space filling curves are the continuous curve which passes through all the closed space without any intersection. There any kinds of space filling curves in present, like Z curve , Gray curve Hilbert curve, but Hilbert curve is employed for this proposed approach because of its superior clustering and its distance preserving properties. A Standard Hilbert Curve (SHC) is used to protect privacy for spatial data. SHC builds indexes of Point Of Interests (POIs) using the same granularity in the spatial domain. If POIs densely distribute, its indexes

generated by SHC will contain a lot of index values without the corresponding POIs; these values are called as null value segments, which is easy for malicious SP to analyze and speculate the distribution of POIs in the transformed space. It will increase the location privacy disclosure risk of the outsourced spatial data. In this proposed system it mentions about three methods for outsourced spatial data and analyzes the security and efficiency of these methods quantitatively. The major contributions are: an index modification method for SHC to improve its security, while a Density-based Space- filling Curve (DSC) is also proposed for efficiency concerns and Hilbert packet list which assign Hilbert index value to each spatial point and then divide the points into packets based on packet size. It is stored in a packet list along with the starting and ending Hilbert index of each packet.

II. PROPOSED APPROACH

The proposed approach [1] applies directly to any location-based service and can be generalized to other domains with ease. Given a spatial database D at the DO with s two-dimensional spatial data points, D = $(d_1,d_2,.....,d_s)$, representing physical locations in the space. The domain of each dataset is normalized to the unit square $[0; 1]^2$ in 2-dimensional Euclidean space, E^2. A spatial range query issued by the AU is bound by two opposite corners of the rectangle $[(cx_0; cy_0); (cx_1; cy_1)]$.
First the Hilbert curve is applied to the spatial data to transform and to unclear the location data points. The data are used by the DO to form Hilbert packet list, where each point stores the all data point along with its Hilbert cell values. Encryption is done to this packet list to hide the information from the SP. Now when the AU requests for any kind of information, the AU converts the range query request to a set of one dimensional Hilbert indices. This integer set is decrypted by the AU with the same decryption key provided to them by the DO.

III. LITERATURE SURVEY

A. Inde x Modification Method

As real-world spatial knowledge make clear to mass distribution, the POI lists of words in a book produced by SHC have within many nothing value parts, which increase the public disclosure. The third party SP has the complete index and may actionlessly come to be some mappings between the POIs and lists of words in a book. By take a look at ever part in turn the lists of words in a book, the attacker can easily discover the unbroken stretch parts and nothing value parts. He can chief place on this knowledge and send in name for his back knowledge to value the placing of unknown POIs.

Before putting into use of index modification method, first let discuss about the null value index for measuring the privacy disclosure risk of the space- filling curves. Null value index are those index values without corresponding POIs. If it is compressed these null value segments, the distribution of the indexes will become equilibrium. So it will be more difficult for the attackers to analyze, and the privacy disclosure risk will decrease. Based on this concept, an index modification algorithm to improve the security of SHC.

1) Null value index
In order to prevent multiple POIs from allocating the same Hilbert value, SHC needs to increase the curve order. As POIs usually distribute in a centralized manner, there may be many null value segments in the generated POI indexes. In this way, malicious SP can analyze these fractal parts of indexes, and then find out the dense areas in transformed space. Although using dummy objects to fill null value segments can explicitly eliminate these segments, malicious SP can discover items with very low query frequency by analyzing query history then verify the ranges of dummy objects and identify null value segments.

2) Index Modification Method algorithm
Algorithm 1
Input: POIs indexes I, max gap N
Output: A modified POIs indexes I*
(1) b = _rst index in I
(2) for each index i _ I do
(3) if i b > N then
(4) i = b + N
(5) I* = I*\(i, encrypted POI)
(6) b = i
(7) end if
(8) end for
(9) return I*

In Algorithm 1[2], based on the POIs indexes I' generated by SHC and user parameter max gap M, which denotes the max gap value between two indexes in the modified POIs indexes I', compare the difference between two neighbour indexes; if the difference is beyond the max gap M, then the latter index should be modified using the forward index and M. The modified POIs indexes I' should be updated with the new tuple, which contains the index i and encrypted POI. The time complexity of this algorithm is O|I'|, where it represents the cardinality of I'.

After applying this algorithm the length of null value segments gets reduced dramatically, and the outsourced spatial datasets are more difficult for the malicious SP to attack.

B. *Density-based space filling curve*

As Index Modification Method starts executing after the indexes have been generated by SHC, index modification method takes more time to generate the POIs indexes. Inspired by the idea that SHC can partition and transform the original space, a new approach is introduced named Density-based Space filling Curve (DSC), which takes the distribution of POIs into consideration. DSC partition the spatial domain according to the capacity, which is the maximum number of POIs a partitioned region contains, denoted as C. It uses fractal rules of Hilbert curve to determine the visiting sequence of each partitioned region [3].

DSC generation mainly goes through two main steps pointed based on capacity, the spatial domain is partitioned by quad tree structure and the generated partitioned regions will be represented as quad tree nodes. The next is based on the curve orientation, starting point, and scaling factor given by DO, each partitioned region is traversed sequentially in accordance with the fractal rules of Hilbert curve, then the sequence number of each partitioned region is generated, and this number is called as DSC value, which is used to build indexes of POIs.

1) *Quad Tree-Based Space Partition*

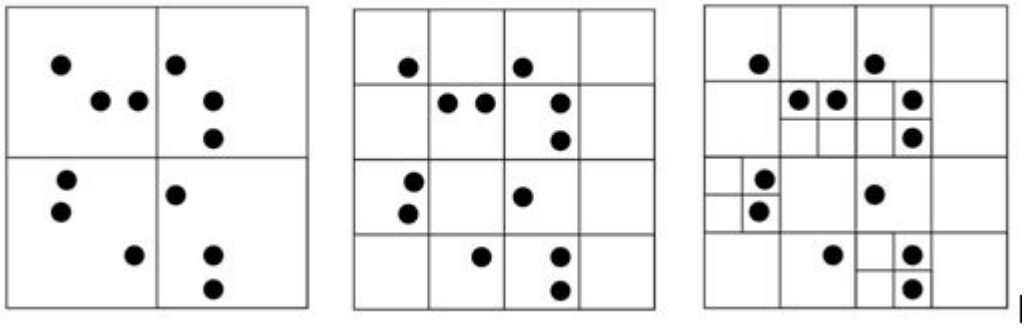

Fig. 1 The quad tree partition: C=3; C=2 ; C=1 respectively[4]

In DSC, the spatial domain is partitioned by quad tree structure. The granularity is decided by capacity C, which means that, under current partition, if the number of POIs in region R exceeds C, it should continue to partition R, until the number of POIs in each partitioned region does not exceed C. Given fig. 1 shows the different partitions using various C. From fig. 1 it seems that the smaller the C is, the more fine-grained the partition is.

DSC partitions the space and generates quad tree nodes corresponding to partitioned regions on the basis of POI dataset and capacity.

2) Index Generation for DSC

After partitioning spatial domain and generating quad tree nodes, it is necessary to generate the leaf nodes DSC value and intermediate nodes sub curve orientation and starting point according to the preset curve orientation and starting point of DSC, as mentioned in Algorithm 2 [3]. Meanwhile, the index value of each POI is set the same as the DSC value of the partitioned region that the POI belongs to. This algorithm employs Hilbert curve fractal rules.

Algorithm 2

Input: quad tree root node M, starting point S_0, curve orientation Θ
Output: updated quad tree root node M
(1) set the orientation and starting point of M as Θ and S_0, respectively;
(2) push M into stack S;
(3) c=0;
(4) while S Θ do
(5) pop the item of S as node F;
(6) if F has child node then
(7) set the orientation and starting point of each child node by F_0s orientation $Θ_0$ and starting point S_0' ;
(8) push child nodes into S according to $Θ_0$ and S_o' by reverse style;
(9) else
(10) set the index value of F as c;
(11) c=c+1;
(12) end if
(13) end while

C. *Hilbert Packet List*

To provide privacy for the spatial data has to be encrypted in both ways. First the space is converted into one dimensional points from two dimensional points using Hilbert Space Key(HSK) and are transformed. The resulting indices and data points

are encoded using the encryption schemes. Keys for encryption and decryption are given by the data owners to authorized users.

1) Hilbert Packet List Construction

Construction of Hilbert packet list at the data owner starts as follows; each spatial point in the space is assigned a Hilbert cell index in the grid. Next data points are stored in packets based on their index value. Each packet is represented as: P_s which is the starting Hilbert index for that packet, P_e is the ending Hilbert index and P_c is a list of a fixed number of original spatial data points in the packet. P_s and P_e represent the location of the first and last data points in the packet, respectively. The P_c size is determined by k, the number of points stored per packet.

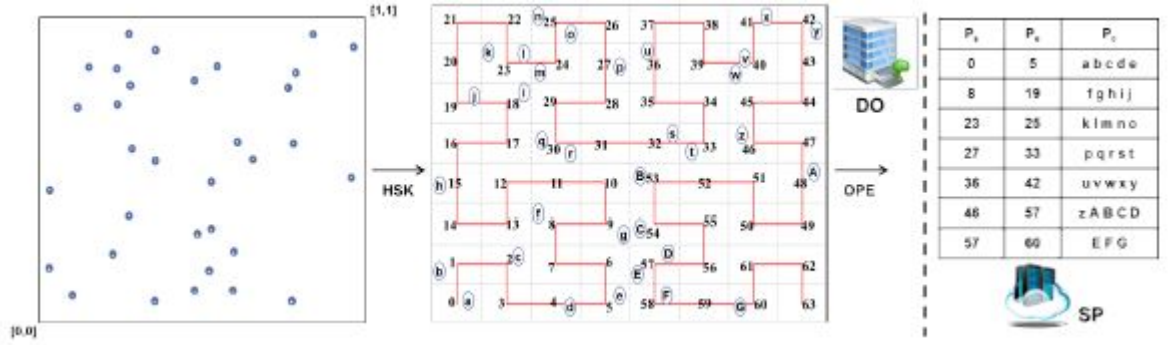

Fig.2 HPL Construction [1]

An empty Hilbert index value indicates that there is no spatial point assigned to it and it will not occur in the HPL unless this index is encountered before the completion of the packet being filled[1]. The rest of the HPL is filled in a similar manner till all the points have been stored. The DO sends the HSK to the AU for mapping the spatial range query to cell indices. To add another layer of privacy to the spatial data, the DO encrypts the HPL before sending it to the SP since it is not a trusted entity. This protects the sensitive data from being leaked by the SP (i.e., to third-party vendors). The SP should not have the ability to decrypt the data. Therefore, it is OPE (Order Preserving Encryption) scheme which is employed here. Additionally, the DO sends the OPE key to the AUs so that it can send encrypted queries to the SP.

Hilbert Packet List Construction Algorithm

Algorithm 3
Input: Hilbert Space Key, HSK
Spatial Data Points, D = (d_1,.....,d_s)
Packet Size, k
Encryption Key, K
Output:
Hilbert Packet List, HPL = (P_1,.....,P_p), where P_i = [P_s; P_e; P_c]

(1) for all d in D do
(2) Normalize d_i
(3) Compute c_x of d_i using F_H and add to C
(4) end for
(5) Sort the set of filled Hilbert cells, C, in ascending order
(6) for all c in C do
(7) $P_s = c_x$
(8) while size(Pc) < k do
(9) Add d_j to P_c
(10) end while
(11) $P_e = c_y$
(12) end for
(13) Encrypt HPL using E_K and send to SP

IV. PERFORMANCE ANALYSIS

A comparison of different schemes used to guarantee that DO and AU can query encrypted spatial data effectively while protecting the location privacy is shown in following table 1. Different features and limitations of each encryption schemes are compared.

The Hilbert-curve does not take the distribution of spatial points into consideration when transforming the original space. In fact, it divides the space using the same granularity and generates Hilbert values to construct the Hilbert index. For resolving this index modification method has been proposed, where null value segments are compressed to improve security. In HPL each spatial point has index value it is divided into packets, along with starting ending Hilbert index value.

By comparing the results shown in the table 1 it becomes more obvious that Hilbert Packet List is the best option for querying spatial data over cloud by ensuring security and efficiency along with less communication cost.

Approach	Security	Efficiency	Cost
IMM	High	Low	High
DSC	Low	High	Low
HPL	High	High	Low

V. CONCLUSION

The amount of outsourcing spatial data are growing everyday and it is very hard to ensure the security and protection to the data along with its locations. For this space filling curves with various methods are introduced and checked against each parameter. As a result of this survey it came to an end that using Hilbert packet list the efficiency and security can be guaranteed, by using range queries the communication cost can also be reduced by a multiple round of communication at a same time. So here it points to a part that among all the methods used mentioned HPL is the best option for the problem in privacy of spatial data.

ACKNOWLEDGMENT

First author- Pursuing post graduation in MES college of engineering Kuttipuram India.

Second author- Working as an Asst. Professor in MES college of engineering Kuttipuram India

REFERENCES

[1] Ayesha M. Talha, Ibrahim Kamel and Zaher Al Aghbari, “Facilitating Secureand Efficient Spatial Query Processing on the Cloud" IEEE Transactions onCloud Computing, Vol. PP, Issue: 99, pp. 2168-7161, July 2017.

[2] Feng Tian ; Xiaolin Gui ; Pan Yang ; Xuejun Zhang ; Jianwei Yang “Security Analysis for Hilbert Curve Based Spatial Data Privacy-Preserving Method",International Conference,June 2014.

[3] Xiaolin Gui ; Jian An ; Pan Yang ; Xuejun Zhang,”A Density-Based Space Filling Curve for Location Privacy-Preserving" Services Computing (SCC), October2014.

[4] Feng Tian ; Xiaolin Gui ; Jian An ; Pan Yang ; Xuejun Zhang, “Protecting Location Privacy for Outsourced Spatial Data in Cloud Storage",Scientific WorldJournal , v.2014; PMC4109601 October 2014.

[5] Ralf Hartmut Gting, "An Introduction to Spatial Database Systems",Special Issue on Spatial Database Systems of the VLDB Journal , Vol. 3, No. 4, October 1994.