Authentication of *k* Nearest Neighbor Queries in the Presence of Obstacles

Sharifa Tahmida Kaniz, Jibon Naher Boishakhi

Introduction

Location-based services:

- Where am I?
- What is here around me?

With the popularity of LBS, the usage of data outsourcing has grown rapidly over the past few years.

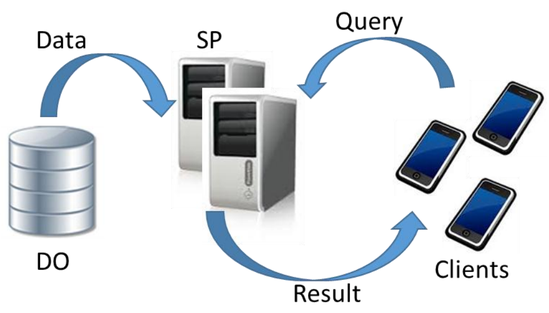
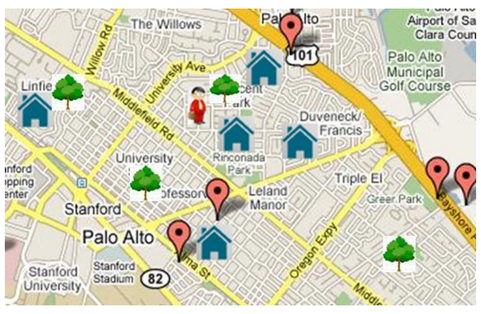


Figure: Data Outsourcing

Motivation

- In data outsourcing paradigm, results may be altered:
  - For personal benefit of SP
  - To reduce computational overhead
  - Hacked by a third party
- How can a client become sure that SP has returned the correct result?
  - Must have some authentication strategies

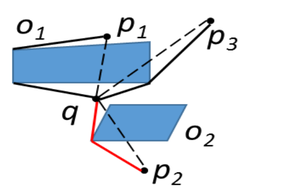
Problem Definition



A pedestrian's path may contain obstacles like buildings, trees or lakes.

q → query point
p1, *p2*, *p3* → *POIs*
o1, *o2* → *obstacles*

A *k*NN query in the obstructed space returns first *k* nearest POIs e.g., restaurants, hospitals or markets that have *k* smallest obstructed distances from the query point *q*.



Our Objective

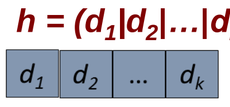
- To authenticate *k*NN queries in the presence of obstacles
  - Existing works[1,2] consider the Euclidean space and road networks
  - No work in the obstructed space

Preliminaries

Our approach based on:

- MR-tree indexing
  - A leaf node contains data objects, d_i
  - An internal node contains MBR_i, hash values, h_i and pointers to child nodes, p_i
- digital signature scheme

$h = (d_1|d_2|\ldots|d_k)$



$h = (MBR_1|h_1|MBR_2|h_2|\ldots|MBR_k|h_k)$

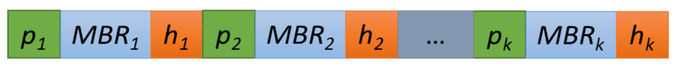
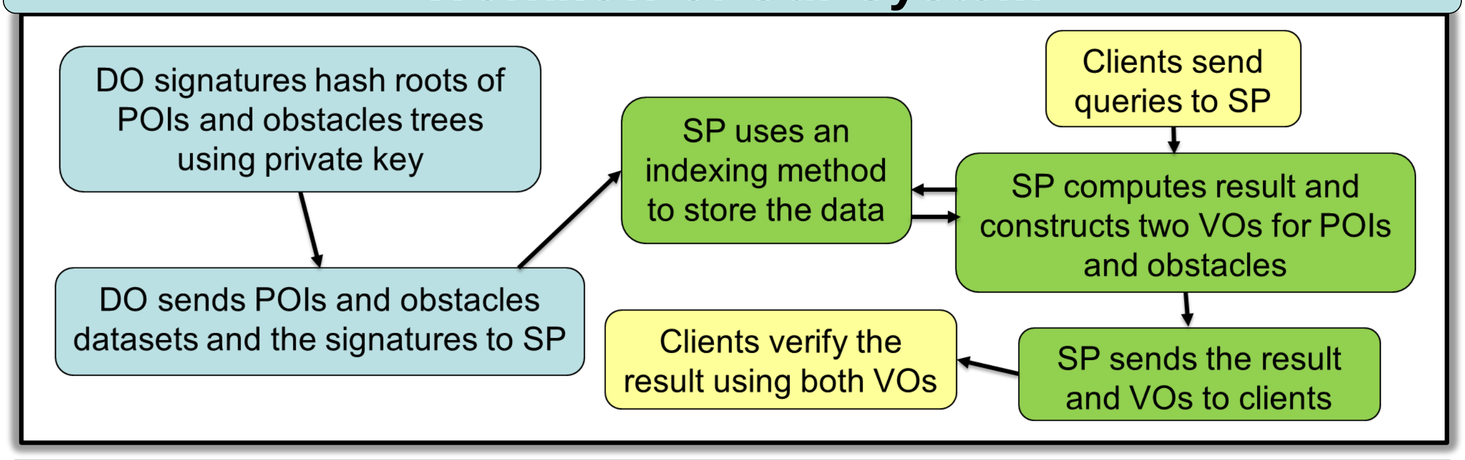


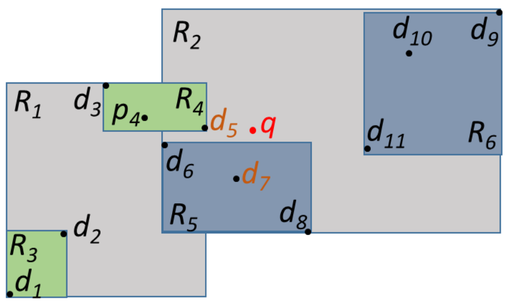
Figure: A leaf node and an internal node of MR-tree

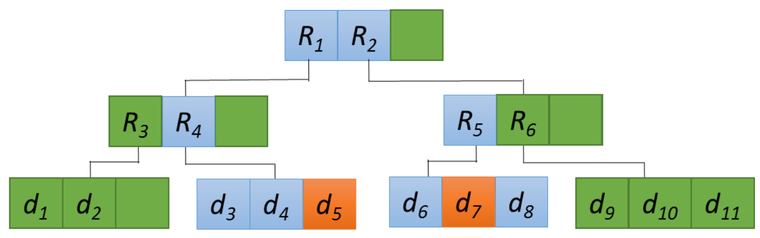
Workflow of Our System



Verification Object Construction by SP

- Two MR-trees:
  - POIs tree
  - Obstacles tree
- Two Verification Objects:
  - VO for POIs
  - VO for Obstacles
- Each VO contains 3 types of entities:
  - Data objects of visited nodes
  - MBR and hash value pairs for pruned nodes
  - Two special symbols to indicate the scope of a node





VO of POIs for the query point *q* :

$$<<<MBR_3,h_3> <d_3,d_4,d_5>><<d_6,d_7,d_8><MBR_6,h_6>>>$$

Verification by Client

Reliability Check:

No POIs or obstacles are added or removed by SP

- Reconstruct hash roots from the VOs
- Decrypt signed hash roots using DO's public key
- Compare these hash roots

Correctness Proof:

Results are the real *k* NNs

- Construct a visibility graph using the extracted POIs and obstacles from VOs
- Find first *k* nearest neighbors using any shortest path algorithm
- Compare the set with result set to detect discrepancy

Evaluation

VO construction in SP: $O(pb^m)$

Verification by client: $O(v + n^2 logn)$

b → fan out of MR-tree
m → maximum depth of tree
v → size of VO
p → number of POIs accessed
n → number of nodes in the visibility graph

Future Challenges

- Implement our approach to evaluate the efficiency and effectiveness
- Develop authentication techniques for range and group nearest neighbor queries

Conclusion

We develop an approach to authenticate *k*NN queries in the presence of obstacles. Our algorithm constructs VO simultaneously with the query processing and ensures that VO contains necessary information to verify the reliability and correctness of the result.

References

1. Y. Yang, S. Papadopoulos, D. Papadias, and G. Kollios, "Spatial outsourcing for locationbased services." in Proceedings of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Canc´un, M´exico, pages 1082–1091, 2008.
2. Y. Jing, L. Hu, W. Ku, and C. Shahabi, "Authentication of k nearest neighbor query on road networks." IEEE Trans. Knowl. Data Eng., 26(6):1494–1506, 2014.
3. J. Zhang, D. Papadias, K. Mouratidis, and M. Zhu, "Spatial queries in the presence of obstacles." in Advances in Database Technology - EDBT 2004, 9th International Conference on Extending Database Technology, Heraklion, Crete, Greece, March 14-18, 2004, Proceedings, pages 366–384, 2004.

Department of Computer Science and Engineering (CSE), BUET