Group Nearest Neighbor Queries over Uncertain Data in Location Based Services

Peng Chen, Junzhong Gu*, Xin Lin and Rong Tan

Department of Computer Science and Technology, East China Normal University,
200241 Shanghai, China
*{pchen, xlin, rtan}@ica.stc.sh.cn; * jzgu@cs.ecnu.edu.cn;*

Abstract

As an important decision support query, Group Nearest Neighbor (GNN) query has received considerable attention from Location Based Service (LBS) research community. Previous works paid much attention to the uncertain data objects (P). Nevertheless, very little work has done to the scenario when query objects (Q) are also uncertain. In this paper, The Range-based Probabilistic Group Nearest Neighbor (in short RP-GNN) query is introduced to draw a comprehensive discussion for this extended scenario. Two novel pruning methods are proposed to improve the performance of RP-GNN. The effectiveness, efficiency and scalability of proposed methods are validated through extensive experiments. The proposed methods achieve an average speed-up of 62.2% against existing probabilistic GNN algorithms and 1-2 orders of magnitude against linear scan.

Keywords: *Range based queries, Probabilistic group nearest neighbor queries, Location based service*

1. Introduction

In recent years, Location Based Services (LBSs) has been flourished with the advances in Internet, GIS and mobile technologies [1]. The ultimate goal of LBSs is to provide its users with timely information at the right place for their decision making [2]. Due to its wide usage in many LBSs, Group Nearest Neighbor (GNN) query has recently gained much attention. A typical scenario of GNN query is to find a facility which minimizes the maximum (minimum or total) travel distance for a group of users. This, in turn, leads to the latest (earliest or total) time that a user (users) will arrive at the facility [3].

In many LBS scenarios, location information becomes uncertain, especially, when privacy concerns, sampling precisions, and network transmission delays are taken into consideration. Previous works of GNN query [3-5] mainly focuses on the scenarios when data objects (P) are uncertain. Nevertheless, very little work has done to the scenario when query objects (Q) are also uncertain. In this paper, The Range-based Probabilistic Group Nearest Neighbor (in short RP-GNN) query is introduced to conduct a comprehensive discussion for this extended scenario. The example of RP-GNN is illustrated in Figure 1. The same as [4, 6, 7], the location of each object is modeled as a so-called uncertain region UR(p_i) which is centered at C_i and has radius r_i. And each uncertain object (data object, p_i, or query object q_i) can locate within the circle with arbitrary distribution. Similarly, the uncertain region of q_i is denoted as UR(q_i). It is centered at Cq_i and has radius qr_i.

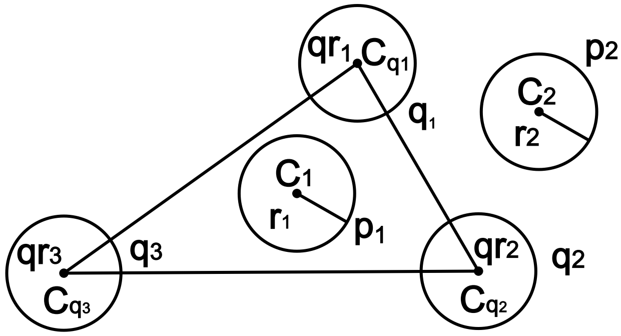

Figure 1. Example of Range-based Probabilistic Group Nearest Neighbor Query

In this paper, two novel pruning methods are proposed to improve the performance of RP-GNN. Query objects pruning (Q_pruning) aims to reduce the number of query objects needed to be considered. And the geometric properties of the RP-GNN problem are exploited in Geometric pruning (G_pruning) method to narrow down the search space. Extensive experiments are conducted to evaluate the effectiveness, efficiency and scalability of the proposed methods under various experiment settings.

The rest of this paper is organized as follows. In Section 2, a brief introduction to related works is given. In Section 3, a formal definition of the problem is presented. Details of the proposed pruning methods and the RP-GNN algorithm are described in Section 4. A systematic performance study is reported in Section 5. Finally, the last section presents the conclusions of this paper.

2. Related Works

Essentially, RP-GNN queries possess the characteristics of GNN queries and range based queries. As such, related works on these two queries are reviewed. The proposed architecture is also outlined in this section.

2.1. Group Nearest Neighbor Query

GNN query as well as its three distance functions (sum, max, min) were first introduced in [5]. As shown in Figure 2, sum is used to minimize the total distance traveled by a group of users, while max (min) can guarantee the latest (earliest) arriving time for a group of users. GNN can be applied in many LBS applications, such as typhoon monitoring, forest fire suppression. Various variants like ANNs [3], GNG [8] and PGNN [4] are proposed subsequently. The RP-GNN query was first introduced by [4], and two algorithm, PSPM (Probabilistic Single Point Method) and PMBM (Probabilistic Minimum Bounding Method), are proposed to cope the RP-GNN problem. However, [4] mainly focuses on the uncertain data objects. It does not pay enough attention to the effect of uncertain query objects. In this paper, we made a more comprehensive study of uncertain query objects under various experiment settings.

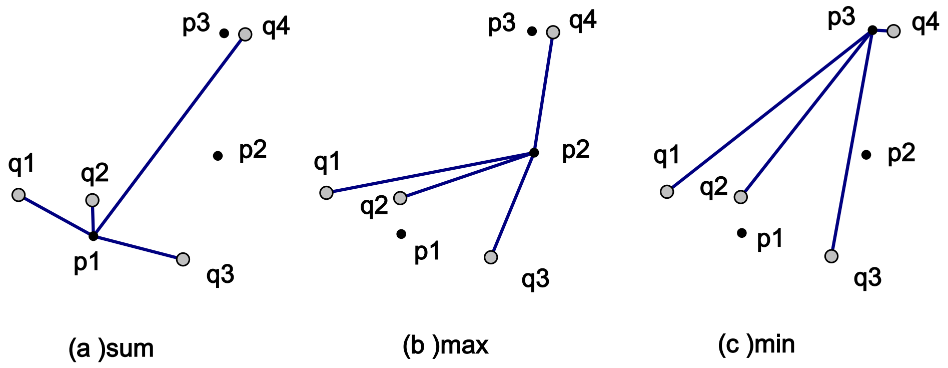

Figure 2. Group Nearest Neighbor Query

2.2. Range based Query

In LBS, Most traditional queries such as Skyline [2, 9] and GNN [5] require exact location information, while this requirement is quite hard to be satisfied in many real world applications. Privacy consideration, sampling precision, and network transmission delay are the three major causes of location uncertainty [2]. The range based query [10, 11] is a better approximation of the real world and concerns more about privacy protection. Those merits make range based query a promising method to solve uncertain location problems. In range based queries, exact locations are replaced by uncertain regions. Accordingly, GNN results are represented by interested locations or items, with probabilities showing the confidences of results [12].

2.3. Proposed Architecture

The architecture of our proposed algorithm is depicted in Figure 3. The experimental system named GaCAM [1], as a successor of LaMOC [13], is a middleware system to support the construction and running of context aware applications.

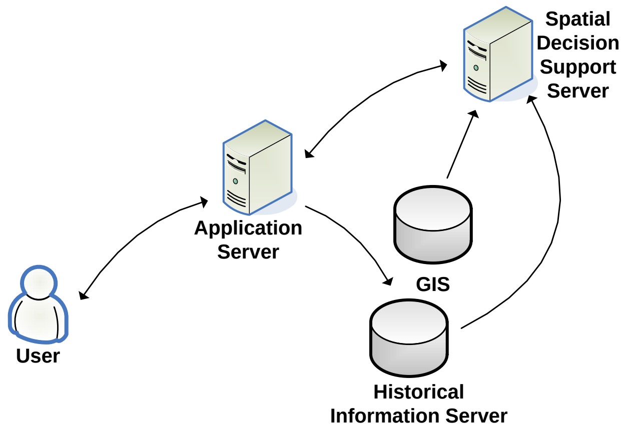

Figure 3. The Architecture of GaCAM

GaCAM has a client-server structure. The client is the application software installed in the mobile device. It is in charge of collecting users' locations, managing the request and data transfer between the client and the server, and displaying the recommended information. The server is composed of the GIS Server, the Spatial Decision Support Server, the Application Server, and the Historical Information Server.

In location information acquiring procedure, users periodically report their locations sampled by cell phones. Location information are then stored in Historical Information Server. The proposed RP-GNN algorithm is implemented as a part of Spatial Decision Support Server. In user request responding procedure, RP-GNN results are combined with users' profiles and GIS information. Finally, personalized information are sent back to users.

3. Problem Definition

For the simplicity of discussion, each UR(q_i) is assumed to be of the same size. When not all UR(q_i) are of the same size, each UR(q_i) can be simply enlarged to have radius max{radius(q_i)}. This will not introduce any false dismissal for RP-GNN results. Although examples are presented in 2D space, our findings still hold in higher dimensional spaces.

For two uncertain object sets P and Q, the RP-GNN query returns object $o \in P$ and probability α. The object o minimizes the maximum distance from o to Q and probability α shows the confidence for this result. Similar to [4], the confidence of RP-GNN result is defined as follows:

$$\alpha = \int_{r_{\min}}^{r_{\max}} (\Pr\{adist(o,Q) = r\} \cdot \prod\nolimits_{\forall p \in P \backslash \{o\}} \Pr\{adist(p,Q) \geq r\}) dr \qquad (1)$$

And adist(o, Q) is defined as the max{dist(o, q_i)}, $i \in \{1 \ldots n\}$.

We can infer from the equation (1) that, a reduction on either Q or P will promote the performance of RP-GNN algorithm. Inspired by this, Q_pruning and G_pruning are proposed respectively to reduce Q and P. The frequently used symbols are summarized in Table 1.

Table 1. Frequently used Symbols

Symbols	Descriptions
P	The data object set with size $\|P\|$.
Q	The query object set with size n.
d	The dimensionality of the data object set.
UR(o)	The uncertain region of object o.
MBR(o)	The Minimum Bounding Rectangle (MBR) of object o.
dist(. , .)	The Euclidean distance between two objects.
adist(o, Q)	The aggregate distance from object o to query object set Q.
LB_adist(o, Q) (UB_adist(o, Q))	The lower (upper) bound of aggregate distance adist(o, Q).
α	The confidence of result.

4. Range based Probabilistic Group Nearest Neighbor Query

In this section, two pruning methods are introduced to improve the performance of RP-GNN query in Section 4.1 and 4.2, respectively. A detailed algorithm and its time complexity analysis is given in Section 4.3.

4.1. Query Objects Pruning

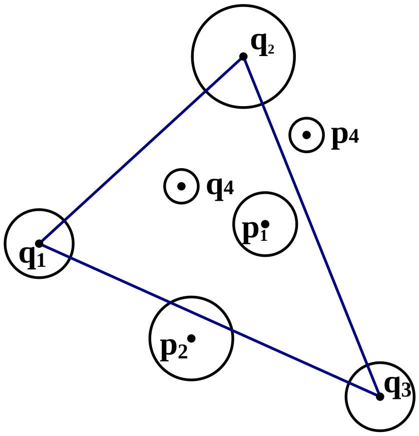

Figure 4. Example of Q_pruning method

At first, let's have a look at the geometric properties of GNN problem. Since for any point in the plane, its farthest point in the set Q must be a point that lies on the convex hull of Q [14]. In Figure 4, q_4 can be safely pruned, since it has no effect on the decision of the convex, and the distance measurements between q_4 and any data object p_i are wasted. Based the aforementioned reasons, only the uncertain query objects which intersect with the convex are considered. Theoretically, Q can be reduced to Q' ($|Q'|>=2$). Since, in the refining phase, the reliability of the result is calculated by taking samples in each q_i. A reduction of query object set will obviously speeds up the refining phase, and thus sharply reduces the overall runtime of algorithm.

4.2. Geometric Pruning

Based on the analysis of the geometric property of GNN problem, we find out that GNN results mostly appear at (or nearby) the center of the query set. This idea is quite similar with traditional SPM (Single Point Method) and MBM (Minimum Bounding Method) [5]. In this paper, the concept named ideal GNN area (I-GNN for short) is introduced to narrow down the search space of RP-GNN algorithm. Here, ideal means without considering data objects.

As depicted in Figure 5(a). At first we made the smallest closing circle of $\mathrm{Cq_i}$ (the center of q_i). It is centered at Op and has radius R. Then two more circles, Cir_1 and Cir_2 are made. Cir_1 is centered at Op and has the radii $R_1 = R - \max\{qr_i\}$. Cir_1 is the smallest circle where, for each q_i, Cir1 $\cap$ UR(q_i) $\neq \emptyset$ holds. Cir_2 also centered at Op and has radii $R_2 = R + \max\{qr_i\}$. Cir_2 covers all UR(q_i).

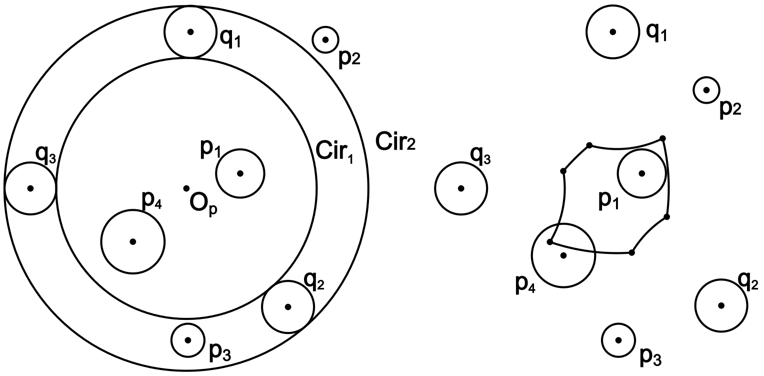

Figure 5(a). Example of G_pruning method and Figure 5(b). I-GNN area

Since dist(I-GNN, q_i) = dist(q_i, I-GNN), and for each q_i, $R_1 <=$ dist(I-GNN, q_i) $<= R_2$ holds. The I-GNN area of Figure 5(a) is depicted in Figure 5(b). It is the intersection of several annuluses. Each annulus is centered at Cq_i with radius R_1 and R_2. The detailed proof has already been reported in our previous work [15] and [16].

When I-GNN contains any data object, data objects outside I-GNN cannot be the RP-GNN result. In Figure 5(b), p_2 and p_3 can be pruned. Otherwise, G_pruning will only increase the complexity of RP-GNN algorithm. It can be seen from extensive experiments that, G_pruning method is effective in most cases and only fails under a few extreme conditions.

4.3. Algorithm

Algorithm 1. RP-GNN(SPM)

/* *P*: a *d*-dimensional uncertain data object set, *Q*: set of *n* uncertain query objects q_1, q_2… q_n, Op: the center of *Q* */

```
I = Rtree(P)// index construction
Q' = Q_pruning(Q)
I = G_pruning(I)// when G_pruning fails, I remains unchanged
get the 1nn of Op in I and initialize best_adist and LB_adist
While(LB_adist< best_adist)
    Get next nearest neighbor knn of Op in I
    If(adist(knn, Q')+radius(knn))< best_adist
        update best_adist and add knn to candidates
    If(LB_adist < dist(knn, Op))
        update LB_adist
end of while
calculate $\alpha$ for each candidate and return refined results
```

Algorithm 1 shows the procedure of PSPM with our proposed pruning method in pseudo code. After the construction of Rtree over the data set (P), the constructed Rtree (I) and uncertain query objects (Q) are pruned by G_pruning and Q_pruning respectively. The nearest neighbor of Op is calculated and then used to initialize best_adist (best aggregate distance so far) and LB_adist. Once the first object with LB_adist >= best_adist has been found, the subsequent ones are pruned. Then all the candidates of RP-GNN are refined by sampling points in uncertain objects and calculating α according to equation (1). Same as [4] and [5], the nearest neighbor algorithm used in our implementation is incremental. The PMBM procedure is quite similar with the PSPM. The RP-GNN algorithm is output sensitive, and its time complexity is $O(ks|Q'|)$, where k is the number of RP-GNN results, s is the number of samples taken in each object in refining phase (Line 12), and $|Q'|$ is the number of query objects needed to be considered.

5. Performance Evaluation

In this section, we evaluated the proposed pruning methods under various experiment settings. Since there is no real data sets available right now, similar to [4, 6, 7], four synthetic data sets (IUrU, IUrG, ISrU, and ISrG) are used. Each data set is in a d-dimensional data space $[0, 1000]^d$. The center and the radius distribution of each data set are shown in Table 2.

Table 2. Distributions of the Data Sets

	Center Ci	radius ri
lUrU	Uniform distribution	Uniform distribution: $r_i=[r_{min}, r_{max}]$.
lUrG	Uniform distribution	Gaussian distribution: mean=$(r_{max}+ r_{min})/2$ and variance $(r_{max}- r_{min})/5$.
lSrU	Skew distribution (skewness = 0.8)	Uniform distribution: $r_i=[r_{min}, r_{max}]$.
lSrG	Skew distribution (skewness = 0.8)	Gaussian distribution: mean=$(r_{max}+ r_{min})/2$ and variance $(r_{max}- r_{min})/5$.

When data sets are generated, they are indexed in Rtrees with 4 Kbytes page size. Each query set is in a $[0, 1000\times\mathrm{MBR}(Q)]^d$ hyper rectangle. All n query object centers and radii are picked randomly and follow the same center and radius distribution as their data object set. For each data object set, $2^d +1$ query object sets are generated. The number of samples per object used in refining phase is 100.

We use two algorithms (PSPM and PMBM) introduced in [4] as the benchmarks. Linear scan method is also chosen as a benchmark. It simply scans all the objects in data object set sequentially and check if equation (1) is satisfied. Same as [4], the query time of linear scan is defined as $(|P|\times d\times 4/4000)^2\times 10$ (assuming 4 bytes per floating number, the 4 Kbytes page size and 10ms per page access). For the default setting (shown in Section 5.1), linear scan takes 81 seconds. All experiments are conducted on a Pentium IV 2-GHz PC with 2-Gbyte memory, and the reported results are the average rumtime of 100 runs.

5.1. Experiment Settings

The experimental parameters are summarized in Table 3. In following sections, in order to study the effect of each parameter, only the chosen parameter changes in its value range, while the other parameters remain unchanged as the default setting.

Table 3. Experimental Parameters

Parameters	Value Ranges	Default Setting
Dimensionality, d	2, 3, 4, 5	3
The size of data set, $\|P\|$	30K, 100K, 300K, 1000K	30K
The radius of the UR(p_i), $r[r_{min}, r_{max}]$	[0, 10] [0, 15] [0, 20] [0, 25]	[0, 15]
The radius of the UR(q_i), $qr[qr_{min}, qr_{max}]$	[0, 10] [0, 15] [0, 20] [0, 25]	[0, 15]
The size of query set, $\|Q\|$	4, 6, 8, 10	6
The size of MBR(Q)	5%, 8%,10%, 15%	10%

5.2. Experiment Results

PSPMQ (PSPMG) and PMBMQ (PMBMG) are the algorithms which only implement Q_pruning (G_pruning) method. In PSPMB and PMBMB, both pruning methods are implemented. The numbers over columns in following figures are the speed-up ratios against linear scan method.

The effectiveness of proposed pruning methods. Figure 6 tests the effectiveness of proposed pruning methods. As shown in figure 6, Q_pruning is more effective than G_pruning, because a reduction of Q can obviously reduce the time cost in refining phase. In ISrU and ISrG, more time are cost, since most data objects concentrate in a small region in skew distributions. High speed-up ratios are achieved over different data sets (by 1-2 orders of magnitude). PSPMB is better than both PSPMQ and PSPMG in most cases. The same trends can also be seen for PMBMB.

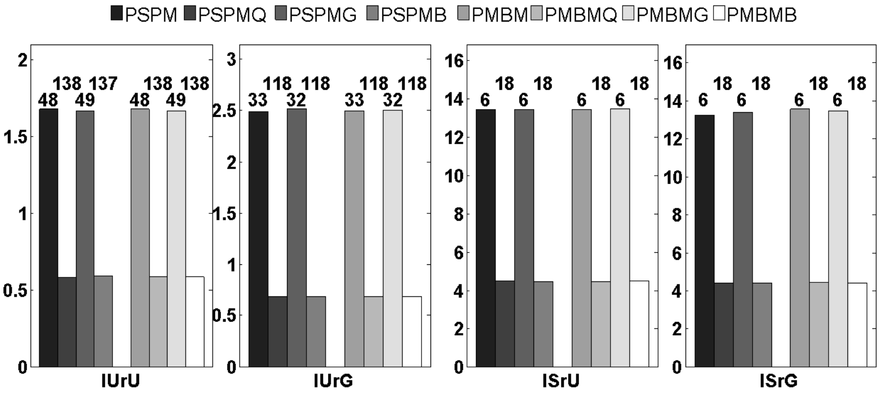

Figure 6. The effectiveness of the proposed pruning methods

Performance versus dimensionality (*d*). The effect of dimensionality is tested in Figure 7. Since the sizes of data object sets are fixed to 30K, a higher dimensionality results in less overlapping. However, a higher dimensionality will also add complexity to each distance calculation. When dimensionality (*d*) is 4, the runtime is the highest. PSPMB and PMBMB are still more efficient than their counterparts. The speed-up ratios of the proposed methods remain high (by 1-2 orders of magnitude).

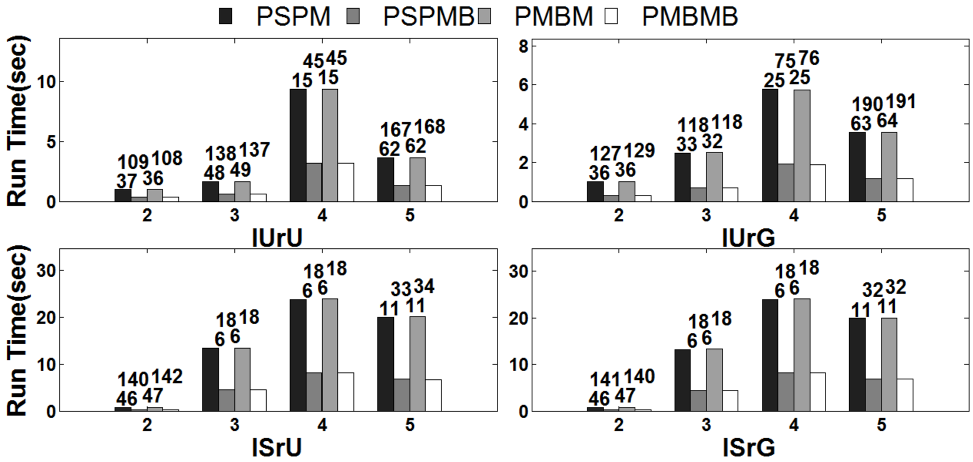

Figure 7. Performance versus dimensionality

Performance versus |*P*|. It can be seen from Figure 8 that, the proposed pruning methods are scalable. The runtime increases linearly with the size of data object set. The high speed-up ratios (by 1-4 orders of magnitude) indicate good scalabilities with respect to the size of data object set.

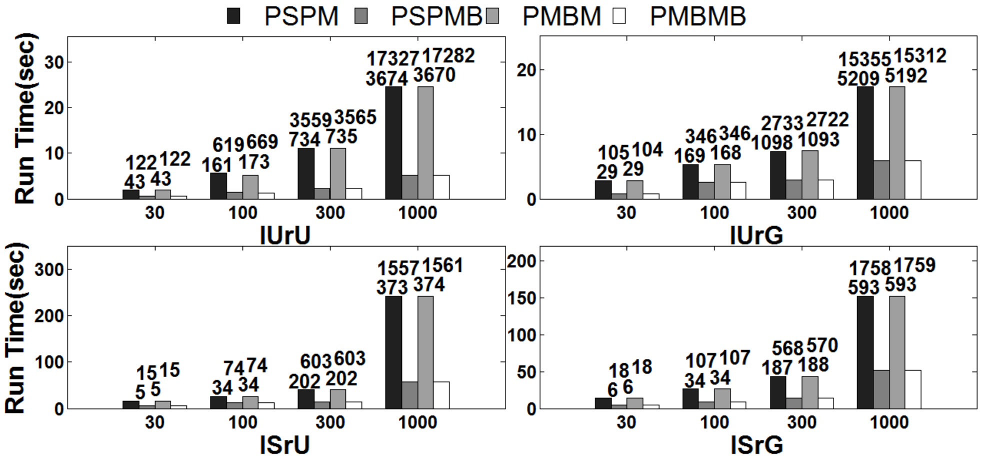

Figure 8. Performance versus |*P*|

Performance versus *r*[r_{min}, r_{max}]. The performance over different values of *r*[r_{min}, r_{max}] is shown in Figure 9. When *r*[r_{min}, r_{max}] becomes large, higher run times is required. PSPMB and PMBMB outperform their counterpart under most conditions. PMPMB achieves better improvements than PSBMB.

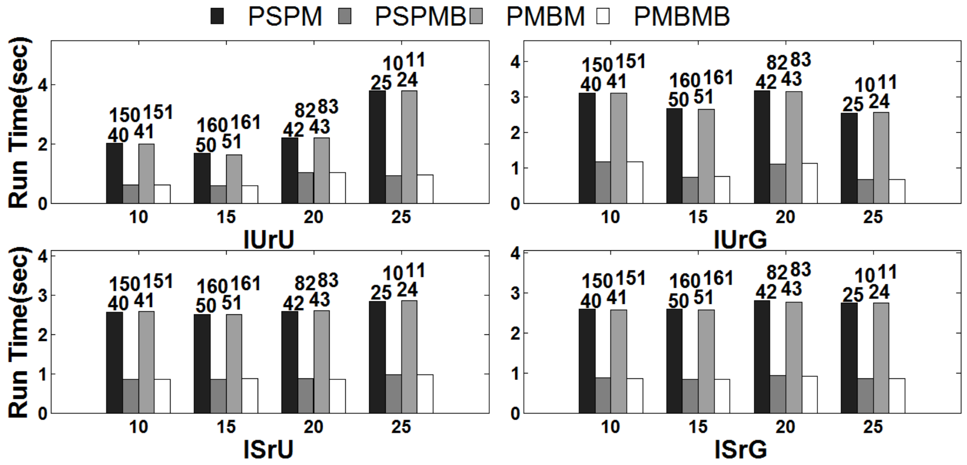

Figure 9. Performance versus $r[r_{min}, r_{max}]$

Performance versus $qr[qr_{min}, qr_{max}]$. The performance under different $qr[qr_{min}, qr_{max}]$ is demonstrated in Figure 10. A bigger *qr* means a higher uncertainty of Q. And a higher uncertainty of Q will make more data objects been recognized as RP-GNN candidates. PMPMB and PSPMB are more efficient than benchmarks in most cases. They also get 1-2 orders of magnitude against linear scan.

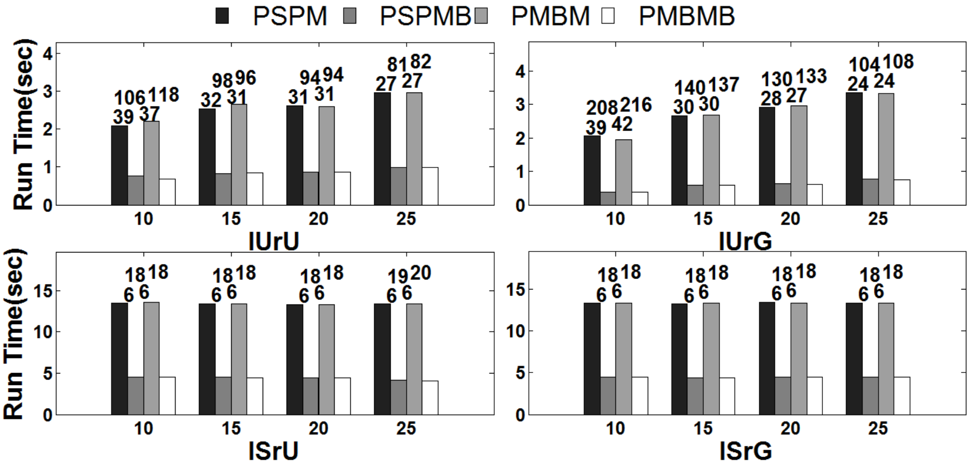

Figure 10. Performance versus $qr[qr_{min}, qr_{max}]$

We also conducted experiments under other parameter settings summarized in Table 3. Although those experiment results are not presented due to space limit, they show similar trends as reported above.

6. Conclusions

In this paper, we conduct a comprehensive discussion on the RP-GNN query. The geometric properties of GNN problem are analyzed and two novel pruning methods are proposed to improve the performance of RP-GNN query. The effectiveness, efficiency and scalability of the proposed pruning methods are validated through extensive experiments under various experiment settings. Experiment results show that the proposed methods outperform PSPM and PMBM by about two-thirds in the default experiment setting. When compared to linear scan, proposed methods have better performance by 1-2 orders of magnitude in most cases.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 60903169, the Shanghai Science and Technology Development Funds under Grant No. 075107006 and 09510703000, and a Shandong Province Young and Middle-Aged Scientists Research Awards Fund with No.BS2010DX012.

References

[1] J. Z. Gu, Communications in Computer and Information Science, vol. 260, **(2011)**, pp. 366.
[2] X. Lin, J. L. Xu and H. B. Hu, IEEE Transactions on Knowledge and Data Engineering, to be published **(2012)**.
[3] D. Papadias, Y. F. Tao, K. Mouratidis and C. K. Hui, ACM Transactions on Database Systems, vol. 2, **(2005)**, pp. 529.
[4] X. Lian and L. Chen, IEEE Transactions on Knowledge and Data Engineering, vol. 6, **(2008)**, pp. 809.
[5] D. Papadias, Q. M. Shen, Y. F. Tao and K. Mouratidis, Proceedings of 20th International Conference on Data Engineering, **(2004)** March 30-April 2; Boston, USA, pp. 301-312.
[6] R. Cheng, D. V. Kalashnikov and S. Prabhakar, IEEE Transactions on Knowledge and Data Engineering, vol. 9, **(2004)**, pp. 1112.
[7] Y. F. Tao, R. Cheng, X. K. Xiao, W. K. Ngai, B. Kao and S. Prabhakar, Proceedings of the 31st international conference on Very large data bases, **(2005)** August 30-September 2;Trondheim, Norway, pp. 922-933.
[8] K. Deng, S. Sadiq, X. F. Zhou, H. Xu, G. P. C. Fung and Y. S. Lu, IEEE Transactions on Knowledge and Data Engineering, vol. 2, **(2012)**, pp. 295.
[9] L. Wang, T. H. Zhou, K. A. Kim, E. J. Cha and K. H. Ryu, International Journal of Software Engineering and Its Applications, vol. 2, **(2012)**, pp. 113.
[10] Y. K. Feng and A. Makinouchi, International Journal of Database Theory and Application, vol. 4, **(2010)**, pp. 1.
[11] J. W. Song, S. H. Shin and S. W. Kim, International Journal of Database Theory and Application, vol. 1, **(2008)**, pp. 29.
[12] L. Antova, C. Koch and D. Olteanu, The International Journal on Very Large Data Bases, vol. 5, **(2009)**, pp. 1021.
[13] J. Z. Gu, L. He and J. Yang, Communications in Computer and Information Science, vol. 61, **(2009)**, pp. 250.
[14] M. D. Berg, O. Cheong, M. V. Kreveld and M. Overmars, Computational Geometry: Algorithms and Applications, 3rd, Springer-Verlag, Berlin Heidelberg, **(2008)**.
[15] P. Chen, J. Z. Gu, X. Lin and R. Tan, Journal of Computational Information Systems, vol. 13, **(2011)**, pp. 4668.
[16] P. Chen, J. Z. Gu, X. Lin and R. Tan, International Journal of Multimedia and Ubiquitous Engineering, vol. 2, **(2012)**, pp. 189.

Authors

Peng Chen received the BS degree from the Department of Computer Science and Technology, East China Normal University, in 2008. He is currently working toward the PhD degree in the Department of Computer Science and Technology, East China Normal University, Shanghai. His research interests include uncertain data, location based service and context aware computing.

Junzhong Gu Professor of Computer Science, Head of Institute of Computer Applications, East China Normal University, China. He received the M.S. degree in Computer Science from East China Normal University in 1982. He works at East China Normal University since 1982. He worked as visit professor at GMD and University Mannheim, Germany (1987-1989, and 1991-1993). His research interests now include context aware computing, distributed data management, and multimedia information processing.