D-034

Generalized Vicinity Query Algorithm in Road Network Distance

Htoo Htoo* **Yutaka Ohsawa***

1.Introduction

In geographic information systems (GIS) and location based services (LBS), queries based on spatial proximity among data objects are important. Various vicinity queries (VQ) on the road network distance such as kNN query, RkNN query, bichromatic reverse nearest neighbor query (BRNN), range query, and high order network Voronoi diagram (NVD) belong to these types of queries. For example in BRNN query, there are two types of data sets, a rival object set (S) and an interest object set (P). When an object q in S is selected, the query retrieves all interest objects in P whose nearest neighbor is q. The simple way for this query is with two steps: (1) making Voronoi region [1] whose generator is q, and (2) finding all data points in P that are included in the Voronoi region. In this paper, this kind of Voronoi region is generated on demand and called vicinity region. Queries in vicinity regions are called vicinity queries.

Fig. 1 shows an example of VQ for BR3NN when k is 3. In this figure, background lines show road network, filled circles show objects belonging to S, white and filled triangles show objects belonging to P. When a query point q in S is specified, BRkNN is defined as follows.

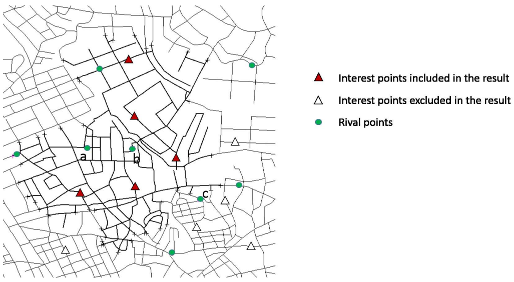


Fig. 1: An example of BR3NN VQ

$$BRkNN = \{p \in P | d_N(p, q) \leq d_N(p, NN_k(p))\}$$

Here, $d_N(a, b)$ shows the road network distance between a and b, and $NN_k(p)$ shows the kth nearest neighbor of p considering only rival objects in S. In this figure, q is set on b. The filled triangles include b in their 3NN, therefore, they are included in BR3NN of b. Contrary, white triangles do not contain b in their 3NN, and hence, they are not members of BR3NN of b.

*Saitama University

Several bichromatic queries similar to VQ have been studied. For example, the region giving the same kNN query result set has been called order-k Voronoi region [1]. In this Voronoi region, the order in the kNN result is not concerned. In this paper, a query that finds all data points in P located in a specified order-k Voronoi region is called a set-kNN query. On the other hand, the Voronoi region in which the order is considered in the result set has been called ordered order-k Voronoi region [1]. This type of query is called an ordered kNN query. If a versatile method for several types of queries is substantiated, it can be applicable to various types of bichromatic vicinity queries.

In these queries, the concept to expand search region is basically implemented by Dijkstra's algorithm. In expansion step, the search region is gradually enlarged in the road network distance starting from a given query point q. In this step, it is also verified whether every adjacent network node to be expanded lies in the same region with the query point at q. In this case, if the verification result is true, adjacent nodes are expanded. When the search region is large or the rival object set S is sparsely distributed, the verifying process in the road network distance takes long processing time. This causes the performance deterioration.

To improve in the efficiency of each query in terms of processing time, this paper proposes an efficient verification method, and introduces an integrated and adaptive framework for several vicinity query methods. Moreover, to evaluate the efficiency of the proposed method, extensive experiments were conducted for set kNN query, ordered kNN query and BRkNN query. Comparing to existing query approaches, the proposed method has a great efficiency in processing time especially when the rival objects are distributed sparsely.

2.Proposed method

The original single source multi-targets A* (SSMTA*) algorithm [2] concurrently finds the shortest paths from a source node to multiple target nodes efficiently. The proposed method is similar to SSMTA*, however, the target points are changed to search sequentially in the proposed method instead of concurrent manner.

In Fig. 2, q is a query point in VQ, and s is a kNN candidate. The A* algorithm is applied to find the road network distance from s to q. The A* algorithm uses a priority queue PQA and a closed set CSA. The

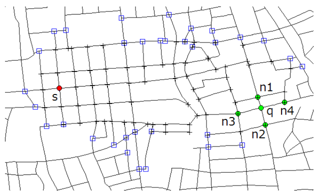


Fig. 2: The distance calculation by SSMTA*

record format in PQA and CSA is the following.

$$< c, v, d > \quad (1)$$

Here, v is a current node in A* search, d shows the distance on road network between s and v. The first item c is the lower bound distance between s and q, that is $c = d_N(s, v) + d_E(v, q)$ where $d_N(s, v)$ is the road network distance between s and v, and $d_E(v, q)$ is the Euclidean distance between v and q.

Then, the region expansion from q to neighboring nodes occurs as shown in Fig. 2. Every time a new neighboring node is investigated, the road network distance from s to the node is calculated verifying the query condition. If A* algorithm is used in this check, almost the same road network nodes are repeatedly processed. To improve the efficiency of the verification, we reuse the contents of PQA and CSA for the verification of the neighboring nodes.

When a new rival object s becomes a candidate of kNN of a node (in the first step, it is q), the distance between s and q is obtained by applying A* algorithm. And then, the contents in PQA and CSA are kept for the next search. When a neighbor node (n) becomes a target, the distance between q and n is obtained by one of the following two cases.

case 1 If n has already been in CSA, the distance between q and n can be obtained by referring d value in the record, that is $n.d$.

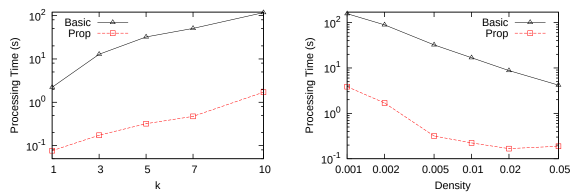
case 2 Otherwise, recalculate c value of all records in PQA for a new target point n, then restart the search by A* algorithm.

In case (1), the search area is not expanded at all. In case (2), this process is necessary because the target node is being changed. By referring updated PQA, the distance search targeting to n is started again. The basic A* algorithm (called pair-wise A* algorithm) finds the shortest path from s to n repeatedly every time the target point is changed. Comparing to it, it is realized that the processing time can be considerably reduced in the improved method. Moreover, the update process for all records in PQA is taken place in the memory and it does not take long processing time.

3.Experimental Results

Extensive experiments were conducted to evaluate the performance of the proposed method by using the real road map in which the number of nodes are 16,284 and links are 24,914. We generated variety of rival object sets and interest object sets on the road network links by pseudorandom sequences. In these experiments, *Density* represents the density of objects and 0.01 for *Density* refers that an object exists once 100 road network links. The proposed method (Prop) and the basic method (Basic) using pair-wise A* algorithm were implemented in Java.

Fig. 3 shows the processing time for BRkNN query (VQ_b). Fig. 3(a) compares the processing time when density is 0.005. The processing time of this query is the most affected by the k value increase. This is because the region size becomes wider rapidly according to the increase of k value. Fig. 3(b) compares the processing time varying the density of objects.



(a) Density = 0.005 (b) $k = 5$

Fig. 3: Processing time of VQ_b

4.Conclusion

In this paper, a generalized vicinity query algorithm in road network distance was proposed. It is an integrated framework adaptable for several vicinity query methods including set kNN query, ordered kNN query, BRkNN query. To improve the performance, modified SSMTA* algorithm was applied for the road network distance verification. With extensive experiments, the proposed method outperformed existing works in terms of processing time. Especially, when the density of distribution of rival objects is sparse, the proposed method is almost 100 times faster than the basic method.

References

[1] A. Okabe, B. Boots, K. Sugihara and S. N. Chiu: "Spatial Tessellations, Concepts and Applications of Voronoi Diagrams, 2nd ed."(2000).

[2] H. Htoo, Y. Ohsawa, N. Sonehara and M. Sakauchi: "Incremental single-source multi target a* algorithm for lbs based on road network distance", IEICE , **E96-D**, 5, pp. 1043–1052 (2013).