A TIME-CONSTRAINED NETWORK VORONOI CONSTRUCTION AND ACCESSIBILITY ANALYSIS IN LOCATION-BASED SERVICE TECHNOLOGY

Wenhao YU, Tinghua Ai*

School of Resource and Environment Sciences,
Wuhan University, Wuhan 430072, China, *tinghua_ai@tom.com

Commission II, WG II/4

ABSTRACT:

Accessibility analysis usually requires special models of spatial location analysis based on some geometric constructions, such as Voronoi diagram (abbreviated to VD). There are many achievements in classic Voronoi model research, however suffering from the following limitations for location-based services (LBS) applications. (1) It is difficult to objectively reflect the actual service areas of facilities by using traditional planar VDs, because human activities in LBS are usually constrained only to the network portion of the planar space. (2) Although some researchers have adopted network distance to construct VDs, their approaches are used in a static environment, where unrealistic measures of shortest path distance based on assumptions about constant travel speeds through the network were often used. (3) Due to the computational complexity of the shortest-path distance calculating, previous researches tend to be very time consuming, especially for large datasets and if multiple runs are required. To solve the above problems, a novel algorithm is developed in this paper. We apply network-based quadrat system and 1-D sequential expansion to find the corresponding subnetwork for each focus. The idea is inspired by the natural phenomenon that water flow extends along certain linear channels until meets others or arrives at the end of route. In order to accommodate the changes in traffic conditions, the length of network-quadrat is set upon the traffic condition of the corresponding street. The method has the advantage over Dijkstra's algorithm in that the time cost is avoided, and replaced with a linear time operation.

1. INTRODUCTION

With the expansion and proliferation of ubiquitous network based communications, mobile services such as location-based services have started to profoundly change the way we make decisions about when, how and where we do things. LBS technology is capable of delivering geographic information and geo-processing power to mobile users according to their current location (Beatty, 2002; Schiller and Voisard, 2004; Li, 2006; Huang and Wu, 2008). Aided by LBS provision and consumption, our capacity has been extended to escape the 'spatial fix' that has constrained in the past our ability to access and consume services outside the home environment.

Location and context are the core elements in LBS which are thereby also called location-aware computing or context-aware services (Schiller and Voisard, 2004). In LBS technology, the spatial-temporal query and accessibility analysis plays an important role including range queries and nearest-neighbor queries in dynamic ways. The LBS providers are inquired most frequently like this: "*What is the nearest gas station to my car?*" or "*How many taxis in a certain area?*". How to establish low-cost, reliable and high-quality services to facilitate user' access to his/her nearest objects of interest (e.g., hotel and restaurant) always is one of the most important issues in LBS applications. To answer these questions, special models of spatial location analysis based on some geometric constructions are required, such as Voronoi diagram. Traditionally, there are two common methods to solve the nearest-neighbor problem. One is the indexing method which has been proved to be inefficient for a dynamic environment. The other is to pre-compute a Voronoi diagram and then answer the closet-facility queries efficiently according to the spatial relationship between the current client position and Voronoi cells for point of interests (e.g. hotels).

Traditionally in Euclidean space, a Voronoi diagram is a partition of the plane into N polygonal regions, each of which is associated with a given point (Lee and Drysdale, 1981; Gold, 1991; Dong, 2008). This kind of VD can be called the planar Voronoi diagram (or P-VD, for short) based on the 2D Euclidean distance. Although there are indeed many cases where the P-VDs can be utilized, the assumption of homogeneous plane becomes inappropriate in other applications, such as the closest-facility queries which need to search the best destination based on the network distance. The associated urban space with the distributed POI facilities is usually not homogeneous and the spatial connection between mobile clients and facilities is actually constrained only to the network portion of the planar space, the so-called network space (Yamada and Thill, 2007). We get the specific type of VD in network space as network VD (N-VD, for short) (Okabe et al., 1992, 2006, 2008), which partitions the street network with the network distance metric used.

By comparing the P-VD, the N-VD is most effective when applied to a micro-scale data set, because the difference between the network distance and the Euclidean distance becomes more significant when dealing with a data set of a finer scale (Okabe et al., 1992; Ratcliffe, 2002; Furuta et al., 2005; Xie et al., 2008; Ai and Yu, 2013). For many years, a number of experts dedicated to the research in this area, and achieve fruitful results.

Okabe et al. (2008) formulated six type of generalized N-VD, including Directed N-VD, weighted N-VD, k^{th} nearest point N-VD, farthest point N-VD, Line N-VD, Polygon N-VD and Point-set N-VD, which have considered the constraints of street directionality, the importance of the feature, and facility data storage type. Their study proposed a computational technique

for constructing the corresponding N-VDs, which is termed as extended shortest-path tree (ESPT). Essentially the ESPT-based method is vector-based, representing the network with set of nodes and segments, and establishing the related network topology with the classic Dijkstra's algorithm. Alternatively, Shiode (2008), Tan et al. (2012), and Xie et al. (2008) leaded another potential field in which they realized the discrete mode of network by dividing real-world network into equal-length linear units, and based on that mode to construct N-VDs and estimate kernel density of traffic accidents occurred on a street network. However, the method by Tan et al. (2012) still calculated the network distance with the vector-based Dijkstra's algorithm, lacking of efficiency for large-scale data sets. On the other hand, their spatial models are developed in a static environment, where constant travel speeds are assumed to be through the network. By utilizing these models to the nearest-neighbor query in LBS, the returned closest facility probably remains the same for each time interval, which may be not in conformity with the reality due to the mobility of mobile clients and changes in traffic conditions (Okabe et al., 2008; Weber et al., 2002).

To better access and consume services that LBS technologies have, this paper presents a N-VD construction method to support both the dynamic spatial query and the accessibility analysis (e.g., dynamic-routing problem, isochrone visualization and range queries) for time-constrained traffic networks. More specifically, we establish a network-based quadrat system and based on that to perform 1-D sequential expansion to generate the corresponding Voronoi subpart for each focus. The idea is actually inspired by the natural phenomenon that water flow extends along certain linear channels until meets others or arrives at the end of route. While the network-based quadrat system will increase the space complexity of the algorithm, the overall efficiency of the alternative algorithm is significantly raised relative to the traditional Dijkstra's ones. The paper is divided into a number of sub-sections including a description of the network-based quadrat and N-VD; a presentation of the basic operator; an overview of the main procedures of the algorithm; an experiment in the Shenzhen city, China; plus a summary of the findings from the study and a discussion of future research directions.

2. NETWORK VORONOI DIAGRAM CONSTRUCTING APPROACH

2.1 Network Quadrat and Description of Network Voronoi Diagram

As a well-known data structure extensively investigated in the domain of computational geometry, the Voronoi diagram defines the neighborhood relations which are more closely related to human perception than other related data models (Gold, 1991,1994). Originally, this construction characterizes spatial proximity in a 2D plane, where the distance of points is determined by their Euclidean distance. Consider a set $\{g_1,...,g_n\}$ containing a limited number of points, called *generator points* or *generators*, in a Euclidean plane. The P-VD is defined as a set of polygons, Vor=$\{Vor_1,...,Vor_n\}$, where the polygon Vor_i is given by

$$Vor_i=\{p| d(p, g_i) \leq d(p, g_j), i \neq j, j=1,...,n \} \tag{1}$$

where $d(p, g_i)$ specifies the minimum distance (e.g. straight line distance) between point p and generator g_i. As for vehicle navigation services, the concerned region often only needs to contain the segments and nodes which are nearest from the generator of interest (e.g., hotel, restaurant), since users are

constrained by the road network— that is, vehicles cannot change direction until reach U-turn locations or intersections. Hence, the Voronoi subpart in network space (termed as *Voronoi subnetwork*) is simplified as partitioning only network paths, instead of the entire plane.

For space partitioning problem, the Voronoi constructions can be computed either in vector mode or in raster mode (Chen et al., 1999). The raster-based method is rather popular due to its simplicity of configuration. Traditionally it involves dividing the study region into a set of regular square quadrats, which are usually represented in the form of a square. While for the cases in network, our method defines a new form of quadrat (i.e. network quadrat) with a unique, linear shape following the irregular configuration of networks, as shown in Figure 1. Within this network-based quadrat system, the unique spatial structure of network which may be lost in the 2D quadrat system, can be maintained by establishing the topological relationship of the linear quadrats. Besides with projecting generators to the nearest quadrats (termed as *generator quadrats*), the Voronoi subnetwork Vor_i can be further defined as

$$Vor_i=\{q| d_L (q, g_i) \leq d_L (q, g_i), i \neq j, j=1,...,n\} \tag{2}$$

where $d_L (q, g_i)$ specifies the network distance between quadrat p and generator g_i in terms of the number of the quadrats connecting their shortest path. Figure 2 shows an example of N-VD within network quadrat system.
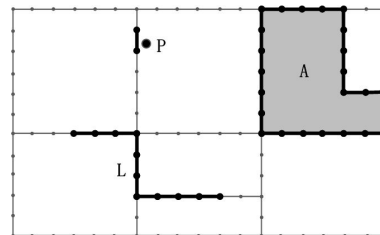

Figure 1. Illustration of a network quadrat system. Each segment is divided into quadrats of a defined network length, and the end points of quadrats are presented as dots. Here, Point generator *P*, Line generator *L* and Area generator *A* is projected on the corresponding *generator quadrats*, depicted as dark quadrats.
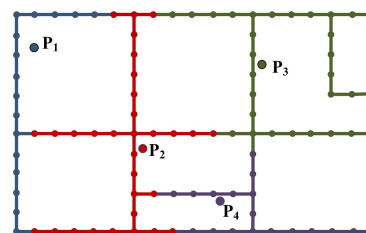

Figure 2. Network Voronoi diagrams within quadrat system (The quadrats showed in different colors belong to the Voronoi subnetworks of the generators with the corresponding colors)

2.2 Process 1-D Sequential Expansion

The location of each quadrat, its adjacency and arrangement among other network quadrats are used for conducting a 1-D sequential expansion operator (or *expansion*, for short) which is the basic operator to construct the N-VDs. As shown in Figure 4, expansion is adjoint of each other and transitive on the network topologies. By progressively processing the expansion with one step moves outward one step, the number of steps from current quadrat to other quadrats can be easily retrieved, and that

provides an alternative way to calculate the travel cost in street network.

To describe the operator more clearly, Figure 3 shows an example with the traditional dilation as comparison. It can be observed that the expansion operator has to take the anisotropic characteristic of network space into consideration, but for dilation operator the adjacency relation can be in the all neighbor four or eight directions. Using expansion operator we can perform accessibility analysis in street network, e.g. to build buffering range based on route distance. Essentially the transmitting in this way is just coincident with the natural phenomena of flow extension in channels, which explains the inspiration for this research.
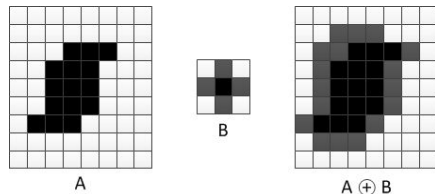

Figure 3. An illustrative example of the dilation operator based on the raster structure of planar space.

(a) Facility points and corresponding generator quadrats

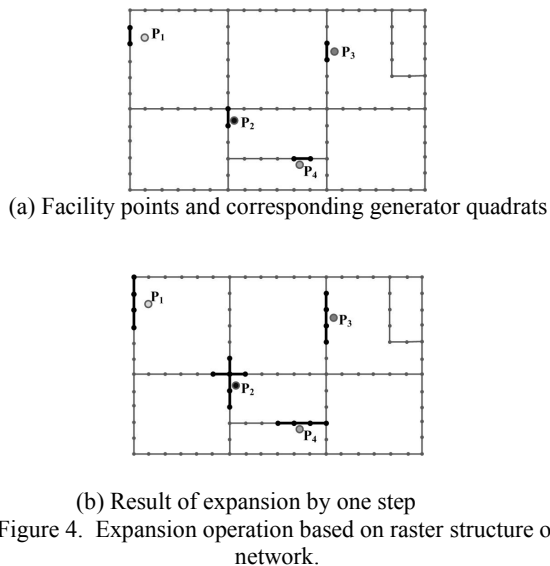

(b) Result of expansion by one step
Figure 4. Expansion operation based on raster structure of network.

It can also be observed that the operator shown in Figure 4 is implemented in a static environment, where the effects of daily congestion are uniform distributed over the road networks and affect all spatial processes equally at all times. However, since the real traffic conditions change over time, the ordinary operator has to be adjusted dynamically to reflect the real-time spatial proximity between facilities.

Local mobility in dynamic conditions is directly reflected in the heterogeneous travel speeds. So to simulate the driver behaviour in real world, we extend the ordinary expansion with controlling the length of quadrat, which is equal to a specified path length depending on the local traffic capacity. The improvement of the path length of quadrat can be defined as:

$$L_B = d \bullet f(x) \qquad (3)$$

where $f(x)$ is modelled as a function (called *weight function*) of the real time x, d is the standard length of quadrat on the street

when the weight value of $f(x)$ is equal to one unit. In this way, the faster travel speed is at time x, the greater length of processing quadrat becomes. As shown in Figure 4, it is assumed that the roads R_2 and R_3 at time x has the minimum local travel speed and we take this value as the reference speed and assign the weight value 1. Then the roads R_2 and R_3 obtain quadrat length 1 unit in segment rasterizing. The corresponding quadrats on road R_1 obtain length of 3 units. By temporal constraint rasterizing, we may find that for the same steps (travel costs) the expansion operation travels different lengths of route at different streets.
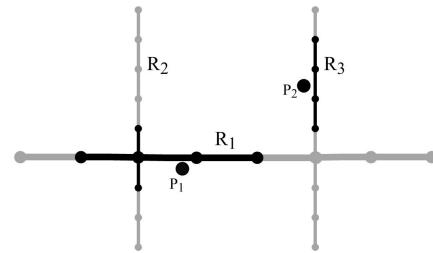

Figure 5. Illustration of expansion operation in a dynamic traffic environment.

2.3 Construction of Time-constrained Network Voronoi Diagram

No matter in the homogeneous space or in the network space, there are three key factors during the process of computing Voronoi diagram by the raster-based operators. The first is a set of appropriate structuring elements for the operation of objects to generate Voronoi subparts, the second is to obtain adjacency relations, and the last is a definition of the end condition. Indeed the structuring element to be used in our approach is a type of network quadrat and the adjacency relation for expansion is actually constrained to the topological relationship of quadrats. When the quadrats occupied by one expansion meet other operators or come to the end of route, the ending condition will be satisfied.

The procedure consists of six main steps, as set out below:
(1) Create a network-based quadrat system out of the original segment set, with dividing each segment into linear quadrats of a defined network length according to the Equation (3).
(2) Within the quadrat system, establish the topological relationship between adjacent quadrats, and quadrats and nodes.
(3) Project the facility points (acting as the generator, also the source of stream flow) onto quadrats, by nearest distance search. Initiate the attribute *cover-state* of each quadrat as unoccupied.
(4) Initiate set *active-set*. For each generator quadrat, if there are neighboring unoccupied quadrat, assign the generator quadrat as active and push into *active-set*, otherwise, assign it *inactive*. For each element in *active-set*, record its attribute *flow-source* the generator ID;
(5) Repeat the expansion for each element in *active-set* until the *active-set* is null:
(6) Generate Voronoi subnetworks by classifying quadrats according to its attribute *flow-source*.

The computational time of the algorithm is mainly in the following steps. First, for dividing the road network into basic linear units, it runs in time of order $O(n)$ where n is the number of road segments. Second, based on the quadrat system, the computational burden in the distance calculating depends on the

number of facility points. If that number is *m*, then this order is dominated by $O(m)$. Hence we realize that the order of the total computational time is dominated by $O(m+n)$, which confirms the hypotheses that the proposed method is more efficient than the vector-based one, namely $O(mn+k \log k)$, where *k* is the number of road intersections (Okabe et al., 2008).

3. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the experiments use a real dataset includes a transportation network and a set of hotel facilities from a central region of Shenzhen, China, as shown in Figure 6. The network possesses 4176 segments, which are classified into three levels: main road, secondary road and ordinary road. According to the speed survey from the traffic administrative department, we set the weight of the corresponding roads as 3.0, 2.0, and 1.0 respectively, which is based on the average traffic speeds of the roads (i.e. 60 km/h, 40 km/h, and 20 km/h). We also choose a certain type of hotel facility for the computation of constrained N-VD and the number is 21.
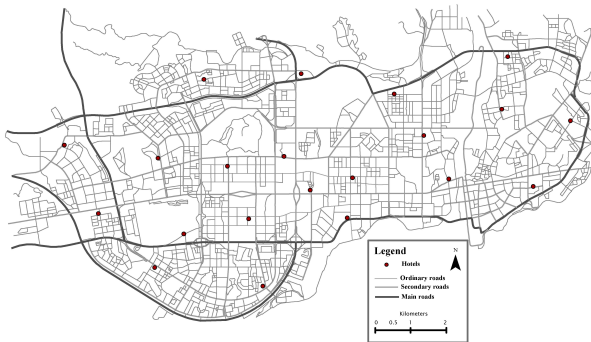

Figure 6. The study area, Shenzhen city, China.

Using the constructing approach mentioned above, Figure 7 captures three key snapshots of the dynamic process with quadrat length at 10m. As the flows extending from the original state to the completion state, it is observed that not all of flows have the same speed. That flows close to the main roads, can reach further places away from the generators in a certain time, while else flows located along minor roads, can only reach nearer places. In fact with temporal constraints added to the practical expansion operator, the flow's speed would largely depend on the local traffic conditions. As the red generator in the northeast part presented, the improvement of traffic environment can make the nearby hotel easy to acquire larger market area than others else.

In addition, Table 1 records the related time cost parameters which are achieved in different lengths of quadrat at 80 m, 40 m, 20 m, 10 m, and 5 m respectively. As quadrat length refines from 80 m to 10 m, the computation times are almost the same for both steps of the pretreatment and composition. However, when the quadrat is assigned to a smaller length, the pretreatment step would occupy the major part of the computational time while the flow extension process only possesses a small proportion of that. It is observed that the time complexity of this algorithm is approximate exponential growth with the refining process of quadrat length. Under conditions which meet the accuracy requirements, one can adjust the split quadrat granularity to efficiently control the performance time.
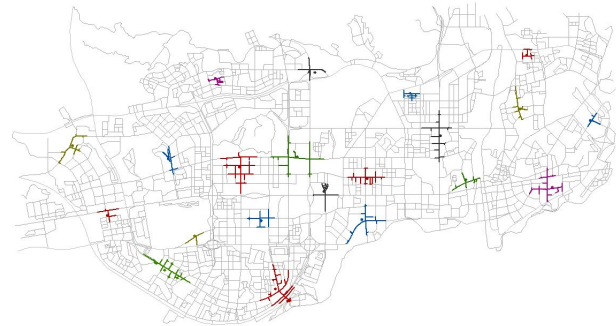
(a) Original state of flow extension.
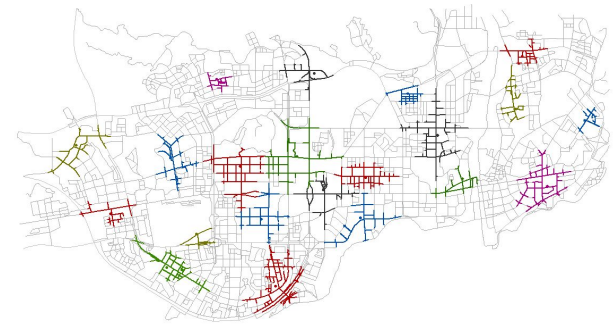
(b) Intermediate state of flow extension.
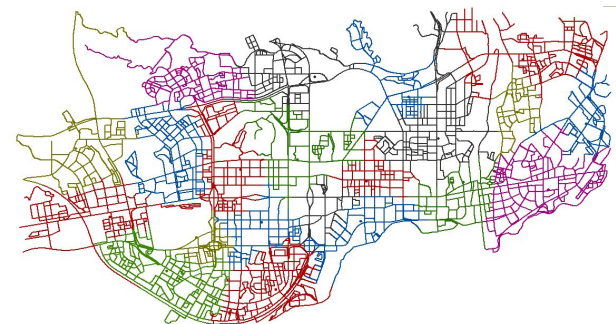
(c) Completion state of flow extension.
Figure 7. Computing generalized network Voronoi diagram of hotel points in Shenzhen.

Table.1 Performance parameter of the algorithm for constructing network Voronoi diagrams

Quadrat length (m)	Number of quadrat in the network	Pretreatment time (s)	Composition time (s)	Total time (s)
80	25259	0.907	0.453	1.360
40	35030	1.015	0.656	1.671
20	57324	1.453	1.000	2.453
10	105170	3.719	1.484	5.203
5	202907	23.500	3.547	27.047

4. CONCLUSION

In this paper, an approach is proposed for generating the constrained N-VDs in a dynamic traffic environment. The approach consists of three main steps: establishing a network quadrat system to represent the linear topological relationship; processing expansion to classify all quadrats; finally, Voronoi subnetworks are achieved by enriching the resultant quadrats in a sequence of consecutive connection. Although we have shown the construction generated by such an approach some issues need to be further addressed in future research. Above all, how

to enrich quadrat-quadrat topologies so that more real constraints (e.g. road direction, turn restriction and facility difference) can be represented and can be dealt with is an advanced question. The flow extension algorithm described in this paper is just used to construct network Voronoi diagrams. The idea could also be used for other accessibility analysis such as isochrone analysis and optimal route selection to facilitate users' trip in mobile context.

REFERENCES

Ai, T., Yu,W., 2013. Algorithm for constructing network Voronoi diagram based on flow extension ideas. *Acta Geodaetica et Cartographica Sinica*. 42(5), pp. 760-766.

Beatty, C., 2002. Location-based services: navigation for the masses,*Journal of Navigation*. 55(2), pp. 241-248.

Chen, J., 1999. A raster-based method for computing Voronoi diagrams of spatial objects using dynamic distance transformation. *International Journal of Geographical Information Science*.13(3), pp. 209-225.

Dong, P., 2008. Generating and updating multiplicatively weighted Voronoi diagrams for point, line and polygon features in GIS.*Computers & Geosciences*. 34(4), pp. 411-421.

Furuta, T., Suzuki, A., Inakawa, K., 2005. The kth nearest network Voronoi diagram and its application to districting problem of ambulance systems. Discussion Paper, No.0501, Center for Management Studies, Nanzan University.

Gold, C.M., 1991. Problems with handling spatial data—the Voronoi approach. *CISM journal*. 45(1), pp. 65-80.

Lee, D.T., Drysdale, R.L., 1981. Generalization of Voronoi diagram in the plane. *SIAM Journal of Computing*. 10, pp. 73-87.

Schiller, J., Voisard, A. (Eds.), 2004. *Location-based services*. Elsevier.

Shiode, S., 2008. Analysis of a Distribution of Point Events Using the Network‐Based Quadrat Method. *Geographical analysis*, 40(4), pp. 380-400.

Li, C., 2006. User preferences, information transactions and location-based services: A study of urban pedestrian wayfinding.*Computers, Environment and Urban Systems*.30(6), pp. 726-740.

Okabe, A., Boots, B., Sugihara, K., 1992. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. John Wiley & Sons.

Okabe, A., Satoh, T., Furuta, T., Suzuki, A., Okano, K., 2008. Generalized network Voronoi diagrams: Concepts, computational methods, and applications. *International Journal of Geographical Information Science*. 22(9), pp. 965-994.

Ratcliffe, J.H., 2002. Aoristic signatures and the spatio-temporal analysis of high volume crime patterns. *Journal of Quantitative Criminology*. 18(1), pp. 23-43.

Tan, Y., Zhao, Y., Wang, Y., 2012. Power Network Voronoi Diagram and Dynamic Construction. *Journal of Networks*. 7(4), pp. 675-682.

Xie, Z., Yan, J., 2008. Kernel density estimation of traffic accidents in a network space. *Computers, Environment and Urban Systems*. 32(5), pp. 396-406.

Yamada, I., Thill, J.C., 2004. Comparison of planar and network K-functions in traffic accident analysis. *Journal of Transport Geography*. 12, pp. 149-158.