A Framework for Multi-Criteria Optimal Location Selection

Ji Zhang[1], Wei-Shinn Ku[1], Xunfei Jiang[2], Xiao Qin[1], Min-Te Sun[3], and Hua Lu[4]

[1]Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36849, USA

[2]Department of Computer Science, Earlham College, Richmond, IN 47374, USA

[3]Department of Computer Science and Information Engineering, National Central University, Taiwan

[4]Department of Computer Science, Aalborg University, Aalborg, Denmark

{*jizhang, weishinn, xqin*}*@auburn.edu* *jiangxu@earlham.edu* *msun@csie.ncu.edu.tw* *luhua@cs.aau.dk*

ABSTRACT

In this demonstration, we present a visualization system offering two advanced solutions that can efficiently address a novel multi-criteria optimal location query by using Overlapping Voronoi Diagrams (OVDs). Our system not only displays an example that applies the advanced solutions to a practical optimal location query, but also visualizes the process of query evaluation by using the solutions, which can help users better understand the two solutions in a more intuitive way. The Real Region as Boundary (RRB) approach calculates real regions of overlapped Voronoi cells, while the Minimum Bounding Rectangle as Boundary (MBRB) approach approximates the cells by their MBRs to simplify region overlapping calculation. For efficient query evaluation and better demonstration, global and local datasets are managed separately and only the local dataset is displayed in a limited visualization area. User interactions are provided in diverse manners in our system by using the Standard Widget Toolkit (SWT) and Open Graphics Library (OpenGL).

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Application—*spatial databases and GIS*

Keywords

Voronoi Diagram, Optimal Location Query, Algorithm

1. INTRODUCTION

An increasing number of optimal location query techniques have been proposed in the literature [2, 3, 4, 5]. Given a set of weighted objects and a spatial region R, the optimal location query returns a location in R with a maximum influence based on the total weighted distance from the location to the given objects [1]. A novel Multi-criteria Optimal Location Query (MOLQ) was proposed in [6], which receives objects in various types and calculates the best location under multiple criteria. These criteria can be represented by type weight and object weight associated with each object. The query result minimizes the total weighted Euclidean distance from the best location to an object in each object type.

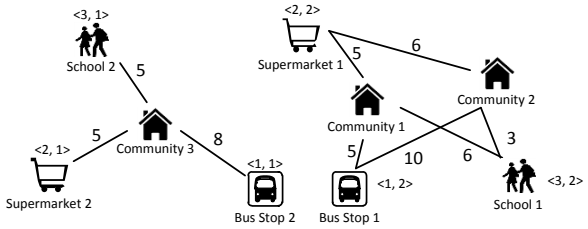


Figure 1: An example of residential location selection. The type and object weights are indicated as $<w^t, w^o>$.

Figure 1 displays a simple example of the novel query for residential location selection in a city. We assume that community developers are planning to find an optimal place that attracts people who prefer living in a new community near a school, a supermarket, and a bus stop. When selecting a community, customers would consider multiple factors, such as the importance of object types (e.g., is the closeness of a school more important than that of a supermarket?) and the preference of objects in a particular type (e.g., is the closest supermarket a Walmart or a Kroger?). These two kinds of considerations are represented by type weights and object weights associated with objects and weight functions in the query. Given two schools, two supermarkets, two bus stops, and three potential community locations in Figure 1, the best place is Community 1, whose total weighted distance (16) is shorter than that of Community 2 (19) and Community 3 (18). The type and object weights are indicated as $<w^t, w^o>$. The weighted distance of a community and an object is calculated as the product of their Euclidean distance and the two weights.

In this paper, we present an advanced framework that visualizes two novel solutions for MOLQ. By exemplifying an optimal location query in a real application and demonstrating the query evaluation of the two solutions, our system can help users understand the solutions in a more intuitive way. Because of the high density of objects, a two-level data management method is employed in our system for efficient data

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.
Copyright is held by the owner/author(s).
SIGSPATIAL'15, November 03-06, 2015, Bellevue, WA, USA
ACM 978-1-4503-3967-4/15/11.
http://dx.doi.org/10.1145/2820783.2820784 .

retrieval and better demonstration. With friendly graphical user interfaces, users can easily interact with our system in diverse manners.

2. SYSTEM ARCHITECTURE

The framework of our demonstration system consists of a data management module, a query evaluation module, and a visualization module. The data management module loads and manages global datasets and the customized visualization configurations, such as the zooming settings and the central location of the visualization area. The evaluation module deals with the design and implementation of two advanced MOLQ solutions and provides query results to the visualization module. The visualization module receives user inputs through keyboard and mouse events, and displays the evaluation processing and query results.

2.1 Data Management

Due to the high density of objects and limited visualization space on screen, a two-level data management method is proposed for efficient query evaluation and better demonstrations. Rather than loading all spatial data from global datasets, we create a logically limited area as the search space for query evaluation and visualization. The area can be selected by users through the graphical user interface by utilizing mouse clicks and zoom-in/zoom-out functions.

2.2 Query Evaluation

Figure 2 illustrates the evaluation framework that receives object datasets, object weight functions, and a type weight function as inputs, and produces an optimal location of the query. In the framework, the query is sequentially evaluated by three steps. In particular, a Voronoi diagram is first generated based on each object dataset and its object weight function. Then, an Overlapping Voronoi Diagram (OVD) is produced by overlapping these Voronoi diagrams. In this step, either the Real Region as Boundary (RRB) or Minimum Bounding Rectangle as Boundary (MBRB) approach is selected by **OVD Overlapper** to complete the overlapping computation. Finally, **Optimizer** finds an optimized location by applying a Fermat-Weber algorithm [6].

2.3 Visualization

The visualization component mainly focuses on user interactions, which receives user inputs and visualizes the evaluation process and query results. Through the graphical user interfaces, users can observe the global datasets and interact with the query evaluation, such as selecting the RRB or MBRB approach, overlapping two or more Voronoi diagrams, and displaying the optimized location and the corresponding spatial objects.

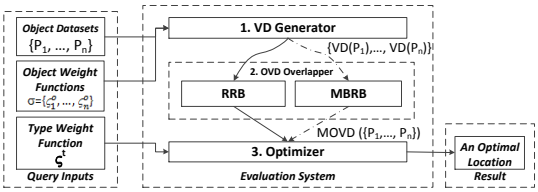
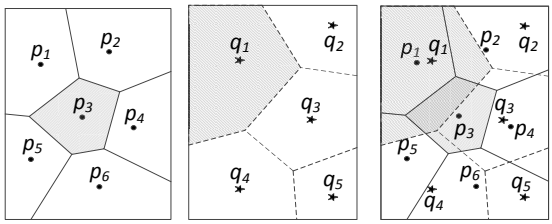


Figure 2: Framework of RRB and MBRB solutions, which are indicated by the solid and dashed arrows, respectively



(a) Schools (b) Supermarkets (c) An OVD

Figure 3: Ordinary Voronoi diagrams and OVDs.

3. MOLQ EVALUATION

We develop two advanced solutions that utilize Overlapped Voronoi Diagrams (OVDs). The RRB approach calculates real regions of Overlapped Voronoi Regions (OVRs), while the MBRB approach abstracts the OVRs by their MBRs to reduce high complexity of region overlapping computation.

3.1 An OVD Example

Before describing our OVD-based solutions, we use a simple example to illustrate the OVD concept. Figures 3(a) and 3(b) display two ordinary Voronoi diagrams generated by schools and supermarkets, respectively. The dominance regions of generators p_3 and q_1 are highlighted by two shaded areas. Figure 3(c) is an overlapped Voronoi diagram that overlaps the two ordinary Voronoi diagrams. The OVD is comprised of a set of Overlapping Voronoi Regions (OVRs), each of which is generated by overlapping two Voronoi polygons. More discussion of OVD can be found in [6].

3.2 RRB Approach

Motivated by plane-sweep-based methods, the RRB approach is designed to efficiently calculate the OVD of two given OVDs. The ordinary Voronoi diagrams can be considered as basic OVDs, the building blocks of the overlap operation. Given two OVDs, the RRB approach sets a sweeping line scanning from top to bottom in the searching space and maintains OVRs that intersect with the sweeping line. For each pair of OVRs intersected with the sweeping line at the same time, the overlapping area of the two OVRs is calculated and produced as an OVR of the output OVD if the OVR is not an empty area. The RRB approach is an output sensitive technique, the complexity of which depends on the number of OVRs in the output OVD. Moreover, the complexity of overlapping two OVRs cannot be estimated until the two input OVDs are determined. Alternative overlapping algorithms may be utilized in the RRB approach due to the variety of Voronoi diagrams. For example, a polygon overlapping method is used in Figure 3(c).

3.3 MBRB Approach

Since the real regions of OVRs will not be used in the optimizer, the MBRB approach that approximates the OVRs by their MBRs is proposed to reduce the potentially high cost of the overlapping calculation. Instead of overlapping the real regions of OVRs, the MBRB approach calculates the MBR of the overlapped OVR by overlapping the MBRs of the two input OVRs. Overlapping two rectangles is much faster than overlapping two polygons or irregular diagrams (e.g., overlapped weighted Voronoi diagrams). However, the MBRB approach inevitably produces false positives. The number of false positives could become larger when overlap-

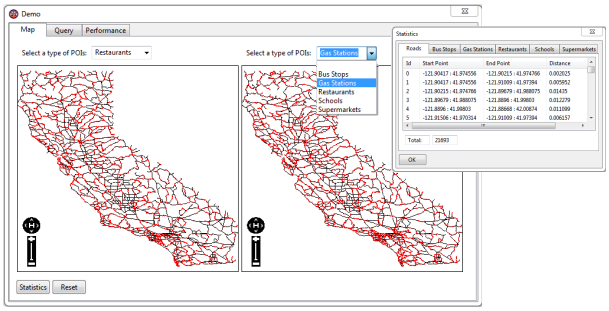


Figure 4: Global datasets.

ping more OVDs.

4. DEMONSTRATION

The RRB and MBRB solutions are implemented in our demonstration system. We use the Standard Widget Toolkit (SWT) and OpenGL library to visualize the graphical framework and user interfaces. The global datasets, the evaluation process of the two solutions, and the performance of the query are displayed in three tab pages, respectively. During the system initialization, all datasets are automatically loaded into the demo system. A spatial object can be represented by a triple <*id*, *coordinates*, *object weight*>, where *coordinates* denote the object location in the search space and *object weight* specifies the importance of the object in the dataset. The type weight of each object type is provided by users in the query evaluation tab page. In the paper, the default values of object and type weights are set to be 1. The query is evaluated in a two-dimensional space.

4.1 Global Dataset Display

Figure 4 shows two examples of two types of objects and California road networks[1]. Five object datasets (bus stops, gas stations, restaurants, schools, and supermarkets) are loaded into our system. The restaurants are plotted in red in the left map, and the gas stations are displayed in the right map. Users can change the type of objects displayed in the map from a drop-down list. From the figure, we observe that the densities of the restaurants and gas stations are high. Demonstrating RRB and MBRB solutions on the global datasets in a limited screen is not user-friendly. Instead, we visualize the solutions with the objects in a smaller area. Moreover, two types of zooming configuration interfaces are supported for better visualization. The map can zoom-in/zoom-out by either scrolling up/down with a mouse, or clicking "+" and "-" symbols in the navigation bar at the bottom left corner of the map. Users can also change the visualization area by either dragging the map in all directions, or clicking navigation arrows. Clicking the symbol "H" in the center of the arrows resets the zooming setting and visualization area to default values. After clicking the "Statistics" button, the statistics information is displayed in a separate window, which is shown on the top right corner of Figure 4. The "Reset" button leads users to clear the datasets displayed in the maps.

4.2 RRB Solution Evaluation Display

The evaluation process that employs RRB and MBRB solutions is displayed in the query tab, as Figure 5-9 show.

[1]http://www.census.gov/geo/www/tiger/



Figure 5: Voronoi diagrams of bus stops and gas stations.

Initially, the system displays two object datasets (bus stops and gas stations) in two maps. Based on the object weights, the Voronoi diagrams are automatically calculated and displayed in different colors. Users can change the dataset in each map by choosing an object type from a drop-down list. The type weight of the selected object type can be specified in a text box. Moreover, users can switch between RRB and MBRB solutions by clicking the radio buttons at the bottom left corner of the query tab.

The "Overlap", "Display Optimum", and "Reset" buttons allow users to overlap the Voronoi diagram in the right map onto the one in the left map, display the optimized location of the query based on the OVD in the left map, and set the query process to the initial stage. The "Display Optimum" button is disabled initially because no overlapping operation has been processed.

After selecting bus stops and gas stations as two input datasets and clicking the "Overlap" button, the Voronoi diagram of gas stations is overlapped onto the Voronoi diagram of bus stops. In our example, the restaurant dataset is selected as the third object type and overlapped on the OVD in the left map, shown in Figure 6. The Voronoi polygons are displayed in different colors (bus stops in red, gas stations in green, and restaurants in blue) in order to easily identify the original Voronoi cells and objects in each OVR. After an overlap operation is completed, the object type selection and type weight input box for the left map are disabled, since they are only used for the object dataset initially displayed in the left map, while the object selection and type weight box above the right map are used for all other object types. The "Display optimum" button is enabled for finding

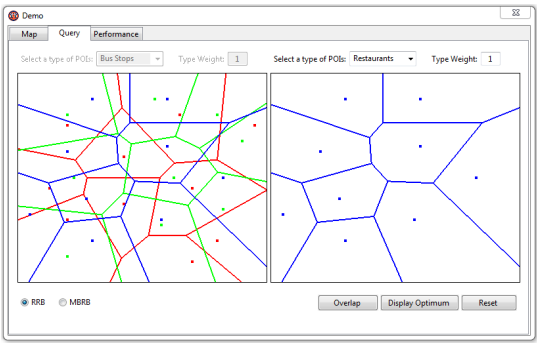


Figure 6: An OVD generated by bus stops, gas stations, and restaurants.

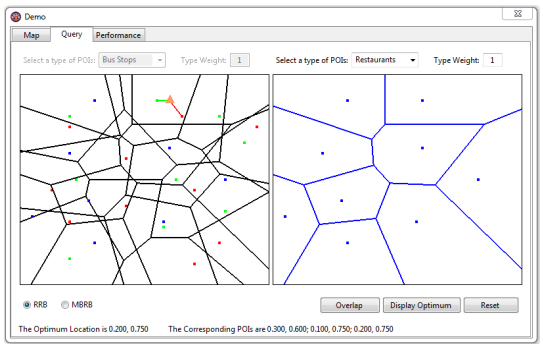


Figure 7: The optimal location.

the global optimal location of the query based on the OVD in the left map.

Figure 7 displays the query result, an optimized location (highlighted by an orange triangle) that minimizes the weighted distance from the location to an object in each type. All polygons are displayed in black, while objects are shown in their own colors. After clicking the "Display optimum" button, the system calculates the Fermat-Weber point of each possible object combination. The globally best location and the corresponding objects are connected; the connection lines are plotted in the objects' colors. In the case of Figure 7, the query result is located on the restaurant object, which indicates that any location closest to the restaurant will minimize the total weighted distance (according to the property of the Fermat-Weber problem in three object type cases, the Fermat-Weber point is the vertex, the angle of which in a triangle is greater than 120 degrees). Moreover, the coordinates of the query result and its corresponding objects are displayed at the bottom of the query tab.

4.3 MBRB Solution Evaluation Display

Differing from the RRB solution, the MBRB solution maintains MBRs of each OVR in an OVD. As shown in Figure 8, when the MBRB radio button is selected, our demo system switches to MBRB mode, in which object datasets and their MBRs of Voronoi cells are displayed in the two maps. The object datasets and their type weights can be specified in the same manner as the RRB solution.

After overlapping bus stops, gas stations, and restaurants, the output OVD is displayed in the left map of Figure 9. The MBRs of Voronoi cells and their objects are highlighted

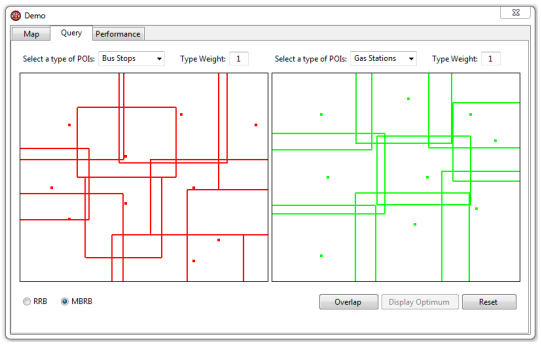


Figure 8: MBRB solution displays MBRs of Voronoi cells in the Voronoi diagrams of bus stops and gas stations.

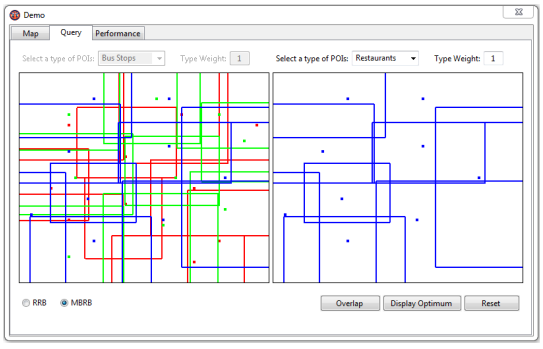


Figure 9: MBRB solution overlaps MBRs instead of real regions of OVRs.

in different colors. In the final step that finds the globally optimal location, the figure is similar to Figure 7 except for the regions of OVRs. The same result will be produced by the MBRB solution. The result and the corresponding objects are displayed at the bottom of the query page. The figure of locating the optimal location is not shown due to the space limit.

4.4 Performance Display

To quantitatively demonstrate the performance of RRB and MBRB solutions, two examples, the numerical results of "three object types" and "four object types" cases, are integrated and presented in the "Performance" tab as shown in Figure 10. Two groups of performance evaluation metrics are supported by varying the number of objects in each type.

5. REFERENCES

[1] Y. Du, D. Zhang, and T. Xia. The Optimal-Location Query. In *SSTD*, pages 163–180, 2005.

[2] Y. Gao, B. Zheng, G. Chen, and Q. Li. Optimal-Location-Selection Query Processing in Spatial Databases. *IEEE Trans. Knowl. Data Eng.*, 21(8):1162–1177, 2009.

[3] J. Qi, R. Zhang, L. Kulik, D. Lin, and Y. Xue. The Min-dist Location Selection Query. In *ICDE*, 2012.

[4] X. Xiao, B. Yao, and F. Li. Optimal location queries in road network databases. In *ICDE*, pages 804–815, 2011.

[5] D. Zhang, Y. Du, T. Xia, and Y. Tao. Progressive Computation of the Min-Dist Optimal-Location Query. In *VLDB*, pages 643–654, 2006.

[6] J. Zhang, W.-S. Ku, M.-T. Sun, X. Qin, and H. Lu. Multi-criteria optimal location query with overlapping voronoi diagrams. In *EDBT*, 2014.

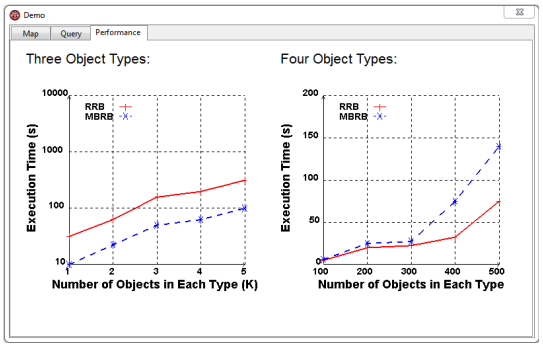


Figure 10: Performance of "three object types" and "four object types" cases.