Broadcast-Based Spatial Queries

Kwang-Jin Park, Moon-Bae Song, and Chong-Sun Hwang

Department of Computer Science and Engineering, Korea University 5-1, Anam-Dong, Seongbuk-Ku, Seoul 136-701, Korea
E-mail: {kipark, mbsong, hwang}@disys.korea.ac.kr
Received December 7, 2004; revised April 29, 2005.

Abstract  Indexing techniques have been developed for wireless data broadcast environments, in order to conserve the scarce power resources of the mobile clients. However, the use of interleaved index segments in a broadcast cycle increases the average access latency for the clients. In this paper, the broadcast-based spatial query processing methods (BSS) are presented for the location-based services. In the BBS, broadcasted data objects are sorted sequentially based on their locations, and the server broadcasts the location dependent data along with an index segment. Then, a sequential prefetching and caching scheme is designed to reduce the query response time. The performance of this scheme is investigated in relation to various environmental variables, such as the distributions of the data objects, the average speed of the clients and the size of the service area.

Keywords  broadcasting, location-aware, caching, prefetching, mobile computing

1  Introduction

Recently, the efficient storage and retrieval of moving objects in Database Management Systems has attracted considerable attention. In Location-Aware Mobile Services (LAMSES), the server is characterized by a large number of mobile clients and stationary objects that they have to manage. In this environment, both mobile and stationary clients have the ability to issue spatial queries. The broadcasting of spatial data is an effective way of disseminating data in a wireless mobile environment, since this method can be scaled up without any penalty being incurred, when the number of users grows. In broadcast-based approach, mobile clients must wait until the server broadcasts the required information. Therefore, client waiting time is determined by the overall length of broadcast data. Moreover, performance of the query processing obtained with a broadcasting method is highly dependent on the order in which the data is broadcasted. Therefore, the question of how to organize the sequence of the broadcast data is a very important issue.

In the broadcast-based model, the broadcasting of data together with an index structure is an effective way of disseminating data in a wireless mobile environment. Using an index can help the client to reduce the amount of time spent listening to the broadcast channel. However, the average time which elapses between the request for the data and its receipt may be increased as a result of these additional messages. Air indexing techniques can be evaluated in terms of the following factors:

- **Access Latency**: the average time elapsed from the moment a client issues a query to the moment when the required data item is received by the client.
- **Tuning Time**: the amount of time spent by a client listening to the channel.

The Access Latency consists of two separate components, namely:

- **Probe Wait**: the average duration for getting to the next index segment. If we assume that the distance between two consecutive index segments is L, then the probe wait is L/2.
- **Beast Wait**: the average duration from the moment the index segment is encountered to the moment when the required data item is downloaded.

The Access Latency is the sum of the Probe Wait and Beast Wait, and these two factors work against each other.

In general, the fastest access time in a broadcast cycle is obtained when there is no index, but this increases the Tuning Time. On the other hand, increasing the number of index segments in a single broadcast cycle reduces the Tuning Time, but increases the Access Latency. Therefore, the number of indices in the broadcast cycle has to be optimized by taking into consideration the trade off between the Tuning Time and the Access Latency. Consequently, in order to achieve efficient indexing on air, it is necessary to simultaneously minimize both the Tuning Time and Access Latency by adjusting the number of indices in the broadcast cycle.

In this paper, we aim to provide research directions towards reducing both the Tuning Time and Access Latency for the NN (Nearest Neighbor)-query and the range-query processing. Let us consider two examples: first, a user drives in his own car and sends a query, such as, “As I am moving in a certain direction, show me all gas stations within 10km of my location.” According to the user’s location and the size of the query range, the result will be dynamically changed. Second, a salesman drives a car and has to visit all of his customers. The salesman sends a query, such as, “What are the names and addresses of the markets near to my current location?”, using his mobile device. Once the salesman gets the answer from the server, he will visit the other markets by the nearest order, and the markets that he visits...
already is going to be excepted in the visiting list. To handle such a query, the positions of the objects and the mobile client must be found. For location-aware range queries, we first introduce the broadcast-based location dependent data delivery scheme (BBS). In this scheme, the server periodically broadcasts reports, which contain the IDs of the data objects (e.g., building names) and their location coordinates, to the clients. These broadcasted data objects are sorted sequentially based on their location before being broadcasted. Then, we introduce a prefetching scheme for use with dynamic range queries in LAMs. The main contributions of our work can be summarized as follows.

- The client can perform range and NN query processing, even if the desired data objects arrive before the associated index segment is received. This technique significantly reduces the latency in broadcast-based location-aware query processing.
- The client simply adjusts the value of \( k \) when performing \( k\text{-}NN \) (nearest neighbor) query processing.
- The client can also perform \( NN \) and range query processing without an index segment. In this case, the best access time is obtained, since no index is broadcast along with the file.

The unlimited mobility of clients makes the access to location-dependent information a new and challenging topic, especially when a client issues a spatial query. The performance of a location-dependent query processing highly depends on the assumed mobility pattern. In this paper, we assume that the client’s mobility pattern follows Random Waypoint Mobility Model.

The remainder of the paper is organized as follows. Section 2 provides background information on the index model and cache maintenance scheme. Section 3 describes the proposed BBS scheme and prefetching method. A performance evaluation is presented in Section 4. Finally, Section 5 concludes this paper.

2 Background

With the advent of high speed wireless networks and portable devices, data requests based on the location of mobile clients have increased in number. However, there are several challenges to be met in the development of Location Aware Query Processing, such as the constraints associated with the mobile environment and the difficulty of taking the user’s movement into account. Hence, various techniques have been proposed to overcome these difficulties.

2.1 Broadcast Model

Disseminating data through a broadcast channel allows simultaneous access by an arbitrary number of mobile users and thus allows efficient usage of scarce bandwidth. In [7], the authors introduce a technique for delivering data objects to the clients in asymmetric environments. In this scheme, groups of pages, such as hot and cold groups, with different broadcast frequencies are multiplexed on the same channel. Then, items stored on the faster disks are broadcast more often than items on the slower disks. However, the wireless broadcast environment is affected by the battery power restrictions of the mobile clients.

2.2 Basic Index Structures

In [8, 9], a linear index structure is proposed based on the Hilbert curve, in order to enable the linear broadcasting of objects in a multi-dimensional space. The Hilbert curve needs to allocate a sufficient number of bits to represent the index values, in order to guarantee that each of the points in the original space has a distinct value. If \( k \) is the number of bits used for a coordinate in the \( i \)-th dimension of the targeted \( m \)-dimensional space and \( n \) is the number of bits assigned to represent the coordinates, then a total of \( \sum_{i=1}^{m} k_i \) bits need to be allocated to represent the coordinates and the expected time for the conversion is \( O(n^2) \). Besides, with this scheme, in order to identify the sequence of the broadcast data items, the clients have to wait until the index segment is arrives, even if the desired data is just in front of them. Thus, this method has the worst possible latency, because the clients have to wait until the beginning of the next index segment.

The R-tree serves as the basis of many later spatial indexing structures. All of these R-tree-based indexes share the basic assumption that spatial objects are approximated by their bounding rectangles before being inserted into the indexes. R-tree-based methods are better supported by random access storages, such as memory and disk, but not the wireless channels. Information is broadcasted based on a pre-defined and it is only available at the moment when it is broadcast. Consequently, backtracking may incur significant access latency. In [2], the authors present a new index structure, called D-tree. Different from the existing approaches, the D-tree neither decomposes nor approximates data regions; rather, it indexes them directly based on the divisions between the regions in order to eliminate backtracking problem. However, this method has the worst latency because clients have to wait until the index segment is arrives, even if the desired data is just in front of them.

2.3 Index on Air

Data broadcasting in a wireless network constitutes an attractive approach in the mobile data environment. However, the wireless broadcast environment is affected by the narrow network bandwidth and the battery power restrictions of the mobile clients. The broadcasting of spatial data together with an index structure is an effective way of disseminating data in a wireless mobile environment. This method allows mobile clients requesting data to tune into a continuous broadcast channel only when spatial data of interest and relevance is available.