

Intelligent Data Receiver Mechanism for Wireless Broadcasting System

Lien-Fa Lin¹ and Yuan-Ko Huang²

Department of Information Communication of Kao Yuan University^{1,2}

lienfa@cc.kyu.edu.tw¹, huangyk@cc.kyu.edu.tw²

ABSTRACT

Broadcast disk technology has become a popular method for data dissemination in wireless information systems. However, factors such as intentional periodic disconnection by the mobile hosts result in further needs for managing access modes to let the mobile hosts procure highest benefit through the disconnections. In this paper, we propose a new broadcast indexing structure, Data Receiver Indicator (DRI), which allows mobile clients to determine the time interval of active and sleep mode for connection according to turnaround time. Our goal is to develop an intelligent data retrieve protocol that allows mobile users to acquire more valuable data with less power consumption within a limited turnaround time. Our performance results show that DRI substantially outperforms the traditional indexes and especially gets 75% saving on waiting time. Therefore, with our approach, users may acquire more valuable data with less power consumption and reduced waiting time which means faster query and response.

KEYWORDS

Intelligent data receiver, mobile information management, database management system, data broadcasting, wireless information system

1. Introduction

Advances in computer software/hardware and wireless communication technologies have led to the rapid development of mobile information systems. In such systems, because of its asymmetry property (i.e., its downlink bandwidth is much greater than the uplink bandwidth), data broadcast is an efficient and most importantly a scalable means to serve for the mobile hosts (MH) [3]. A well-known and widely discussed method is the so-called *Broadcast Disks* approach [11, 17]. *Broadcast Disks* approach can be implemented in various wireless systems such as Wi-Fi (802.11) and 3G global communication networks.

Although these technologies vary in many aspects, they can be viewed as an abstract cellular model where mobile hosts access information through access points. The abstract model consists of a base station, a number of clients, and a number of channels. A mobile host can use an uplink channel to send a request to the base station and receive the result from a downlink channel. We use an example to illustrate how the broadcast disk technique works.

Example 1: A database of 11 pages is to be constructed into a 3-disk broadcast program. Figure 1 shows the database pages and the process of producing the 3-disk program.

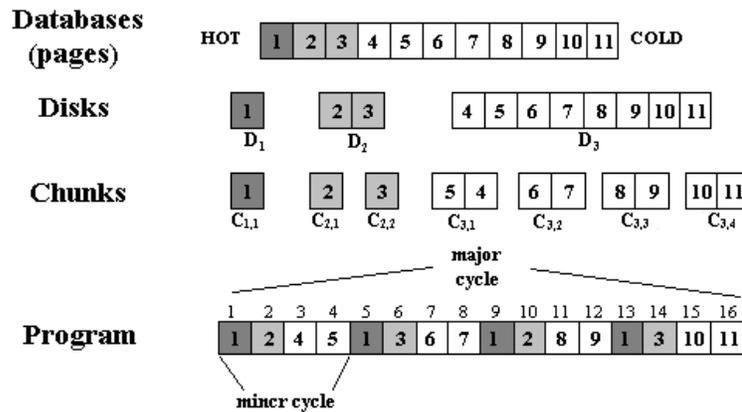


Figure 1: An example of a broadcast disk program.

Assume that data of disk D₁ are to be broadcast twice as frequently as the pages of disk D₂, and four times as frequently as the pages of disk D₃. The 11 pages are divided into three groups (one-page, two-page, and eight-page groups) and each group is assigned to a disk. Also, from the relative broadcast frequency, each disk can be split into a number of chunks. Finally, the broadcast program can be created by taking in turn one chunk of data at a time from the disks. When a chunk from all three disks is fetched, a *minor cycle* is formed in the broadcast program, such as the one indicated in Figure 1. When all chunks of data in the disks are accessed at least once, a *major cycle* is complete.

The main idea of broadcast disks is to broadcast the data of high access probability more frequently than those of low access probability so that MHs would on average spend less time on accessing their required data pages. This broadcast technique can be used in many real-world applications, such as Podcast, an iPod broadcasting mobile classroom promoted by Apple, cell broadcast system [5] in GSM environment, Advanced Traffic Information System (ATIS) [21], wireless classrooms [16], AirMedia Live Internet Broadcast Networks [2], etc. There is also research extending the technique to handle concurrency control of transaction processing to make it a more real methodology for services in wireless information systems [9]. All of the examples of application mentioned above suggest that broadcast disk indeed plays an important role in the broadcast driven real world. Direct Band Network [12] for Smart Personal Objects Technology (SPOT) announced by Microsoft in 2003 in an international consumer electronic exhibition further confirmed that enterprises were interested in providing information service that is available anytime, anywhere with convenient wireless broadcasting.

Research on the design of such a program and its variants are abundant and successful [11]. However, they simply assume that a MH will continuously listen to the broadcast for a long enough period of time to receive the needed information. A new requirement of such a system is that a MH during accessing data is allowed to voluntarily go into doze mode (or disconnect itself), either periodically or nonperiodically, from the server to save its battery life or avoid high wireless communication prices [10].

The focus of this paper is to help mobile hosts who are unaware of the content of the broadcast information and require assistance from the server. We do not address the problem of finding some specific interesting data for a mobile user. There are various occasions that a MH may want to access hot data and skip the cold ones. For instance, while a traveler visits a foreign city and is

listening to the broadcast of recreational service information, he/she may not have any idea where a hot recreational area is. Certainly, this traveler does not know the content of a broadcast program of the local base station either. Hence, a good accessing method for the broadcast system is to tell the user the best combination of connection-time/disconnection-time so that it is the most beneficial (to be defined shortly) to the user if this accessing rule is followed. Current data broadcast schemes do not consider such needs for mobile hosts.

An effective access method will indicate a mobile device when to establish connection to access a large amount of data and when to disconnect. There exists a method of data broadcasting that uses little power to access data, which utilizes index to indicate a mobile device when to wake up and receive the data requested. However, it does not solve the above mentioned problem. The main reason is that the indexed approach [1,15] can only support for those users (or applications) that know what exactly they are going to fetch. The index will guide them to the slot of the required data. In the broadcast disk environment, however, the users need not have such an idea and may not be able to specifically name the data items they required. That is, a mobile user does not know what the content on the broadcast server is in advance. In such situations index cannot help to ease the problem. In addition, some smart applications (e.g. prediction function) do not require users to specifically point out the data to be retrieved but the application must retrieve some data for the prediction function to realize the convenient functions for users, and therefore index is unable to provide a very effective method of access.

Owing to the nature of the environment and the mobile devices themselves, data retrieval in a wireless information system has the following characteristics:(1) storage capacity of a mobile device is limited and unable to store large amounts of data; (2) mobile users possess mobility. After a period of time they may switch to the service coverage of another base station in which the data or the broadcast program schedule may differ from the previous base station, causing a problem for the mobile user to retrieve the needed data and forcing the user to complete data access within a limited time; and (3) a mobile device has a limited power supply and a constant connection normally costs a higher fee to download data which is neither ideal nor practical. Therefore, the design of a highly effective data broadcast system for a wireless environment must differentiate itself from traditional design and give consideration to the features of power saving and a mobile user's limited time for accessing data. Note that in this paper we consider connection/disconnection of a MH as another interpretation of active/doze modes, respectively, in a broad sense. Hence, disconnection of a MH in this paper may not refer to a power-off of the device and complete "disconnection" from the wireless communication network. It may simply be in a doze mode.

In the past, researches on the subject have discussed the structure of data and index for the data broadcast environment [4, 15, 18]. These results provide an effective method of access for single and specific data to be captured. In this paper, we take it further from the past discussion of one specific piece of data to the data on a broadcast server where the content is unknown to mobile users who must access the entire needed information with a method that saves power within a limited turnaround time. The results of this study not only broaden the scope of the past researches, but also provide a more realistic and comprehensive model of wireless application development. Our contributions are four-fold:

- *A new active data receiver model for wireless applications is proposed.*
- *An effective supportive access system that makes the best use of power reserve on a mobile device is designed.*
- *The near optimized computation algorithms are proposed for the most effective data access.*

- *A simulation is conducted to compare with state-of-the-art indexes.*

The rest of this paper is organized as follows. A brief review of the researches related to this paper is provided in Section 2. In Section 3, we describe the supporting data structure and the model. In Section 4, we formally formulate our problem. Then four proposed approaches are presented in Section 5. The performance evaluation is given in Section 6. Finally, we conclude the paper and describe the future work in Section 7.

2. Related Work

Weiser et al. [14] proposed several features of the mobile computing environment and database related topics. Among these, data dissemination is relatively new and seldom discussed in depth. Tomasz Imielinski et al. [22] pointed out that in a wireless network bandwidth of downloading data is larger than that of uploading data due to hardware limitations. In such an environment, it is preferable that the client should not request often so as to save the bandwidth of uploading data. To achieve this, experts suggest the method of transmitting data via broadcast. That is, a server constantly broadcasts data through the broadcast channel using large bandwidth for download. When the client requests data, the request is not sent immediately. Instead, the client monitors the broadcast channel for the desired data. The request is sent to server only when the desired data is confirmed to be not in the broadcast channel. With this method of data broadcast, multiple users may obtain the desired data directly from the broadcast bandwidth without wasting any bandwidth of uploading data, and thus saves bandwidth by a large margin.

This method of broadcast, however, may cause a dramatic increase in users' access time. To reduce this problem, Swarup Acharya et al. [17] proposed broadcast disk that effectively reduced access time. The basic idea of this approach is to increase the broadcast frequency of popular data and reduce the broadcast frequency of unpopular data. Broadcast disk is a simple yet effective method that has formed the basis of many extended researches ever since. These researches may be generally divided into (1) designing mechanism on the client side to further reduce access time, and (2) reducing the power consumption of the client side. The discussions are as follows:

(1) Designing mechanism on the client side to further reduce access time

This type of approach mainly relies on caching [8], prefetching [19], data compression [6] or balanced (push and pull) bandwidth [18]. Swarup Acharya et al. proposed a cache method, called P/X, to reduce access time. P/X selects and deletes either a least requested data or a piece of data that is frequently accessed (but will soon be broadcast again) from the cache according to the data access probability and data broadcast frequency, making space to improve data efficiency. In 1997, Swarup Acharya et al. [18] proposed a backchannel to balance the usage between push channels and pull channels. Quinglong Hu et al. [13], on the other hand, combined cache and pull-based point-to-point data transmission to realize reduction of access time

(2) Reducing the power consumption of the client side

This type of approach uses index techniques in broadcast program to allow mobile users to read and access data with lower power consumption by reducing tuning time [13] spent on reading data at the mobile client's side. This type of research focuses on allowing a mobile device to switch to doze mode when the data being broadcast is not what the user is interested in and switch to active mode when the interesting data is on the air.

Despite the fact that these researches are conducted with the premise of power saving, they are different with our work in the following points: (1) index techniques do not take personal optimization into consideration. What index technology focuses on is reducing average tuning

time instead of optimization for individual users. Different users tune to the broadcast channel at different points in time with different waiting times for index data, causing large differences in data access efficiency. (2) Tuning time does not reflect the real power consumption. Although active mode consumes much more power than doze mode, approximately 20 times more [13], saving tuning time at the cost of waiting time shall not make for the least total power consumption. (3) The environments in which the two types of researches are applied are different. The index discussed in the related literature focuses on effectively processing one specific data, while our study focuses on allowing the client to retrieve enough important data within a fixed turnaround time with the least power consumption.

3. Supporting Data Structure and The Model

In data broadcast applications, when a mobile user request occurs, the mobile client does not send the request directly to the server but monitors for the desired data in the broadcast program. However, the mobile client is usually unable to know all of the content in the broadcast program; the mobile client is unable to determine an effective access mode by itself. In order to conquer this problem, we have come up with a solution that asks the broadcast server to calculate the probable time when the mobile client may start monitoring and the possible length of time of monitoring and add some information in every broadcast program to guide the client to the most effective access mode. To achieve this objective, we need to add some information that we call data receiver indicator, or DRI, to the broadcast data. The data structure of DRI is illustrated in Figure 4. DRI consists of two attributes. The first attribute is turnaround time, T , and the second is the corresponding optimal access mode expressed in the form of a pair (A_i, D_i) . Before listening to the broadcast program, the MH should have an idea in his mind for how long he will listen to the program (i.e., T should be predetermined). When the first data page is received at S_i , the MH gets the DRI of this data page and finds the associated access mode via the column of T values. The MH then follows the suggested access mode to retrieve the broadcast data for the whole T . For example, an MH has totally 8 units of time (i.e., $T=8$) to receive the broadcast data and has now received the first data page in the program. The DRI suggests that $(2, 3)$ be the best access mode for a $T=8$ at this S_i . So, for the rest of the 8 time slots the MH will achieve the greatest *benefit* if it receives data for the next 2 time slots and then skips the next 3 time slots, and so on. If the MH starts to listen at the time that page 8 is broadcast (i.e., a different S_i), however, then $(2,4)$ will be the best access mode. Different start time of the MH and different T may render a different (A_i, D_i) pair. So the computation for searching for the optimal pair is very complicated and costly. The Heuristic Algorithms proposed by Lin et al. [7] can solve this problem by largely reducing the cost of calculation and obtain a near optimal result. However, in order to simplify the task for an easier research, the access mode of such algorithms is fixed. That is, the time a mobile user switches to active mode to receive data, or A_i , is fixed, and the time the same mobile user switches to doze mode to save power and skip unpopular data, or D_i , is also fixed. There is a shortfall in this fixed access mode: when a mobile user sets a longer working time, the fixed access mode will retrieve repetitive data, causing the efficiency of DRI for data access to reduce progressively.

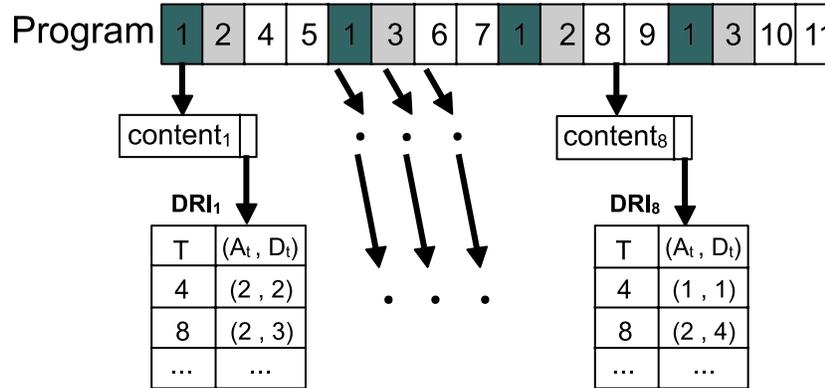


Figure 4: Single Data Receiver Indicator

In this work we have loosened up the limitation of the fixed access mode and allowed mobile users to have several access modes, as illustrated in Figure 5. Mobile users' behavior can be adjusted according to the circumstances of the data broadcasting in broadcast program. By doing so, mobile users can further avoid retrieving repetitive data and make more effective use of limited power. In order to support multiple access modes for mobile users, several necessary changes are made in the DRI table. As illustrated in Figure 5, the access mode has been changed to a new one: (A_t, D_t, cnt) , which represents that after the mobile user continues to access for cnt times, as the efficiency of (A_t, D_t) is reduced progressively the access mode (A_t, D_t, cnt) changes to another set. Such change of access mode shall continue until the working time T ends. Taking Figure 5 as an example, S_t is the time when a mobile user tunes to the broadcast channel and begins to monitor, where $S_t = 8$ and $T = 4$, and the mobile user makes a query for the T in DRI table from the received data page. The suggested access mode for the mobile user is a collection of multiple access modes: $(1, 1, 1)$, $(2, 3, 1)$, and $(1, 0, 1)$. Therefore, the mobile user will adopt $(1, 1, 1)$ mode for one time, followed by $(2, 3, 1)$ mode for one time, and finally change to $(1, 0, 1)$ mode for one time, then the popular data can be retrieved with the most power saving method. In DRI table, T does not have to thoroughly list all possible situations to save DRI capacity. The cost for the saving of DRI capacity is that there are chances that the mobile client may not receive the suggestion of the most effective (A_t, D_t) . If a user's T is not listed in DRI, then the nearest T will be selected.

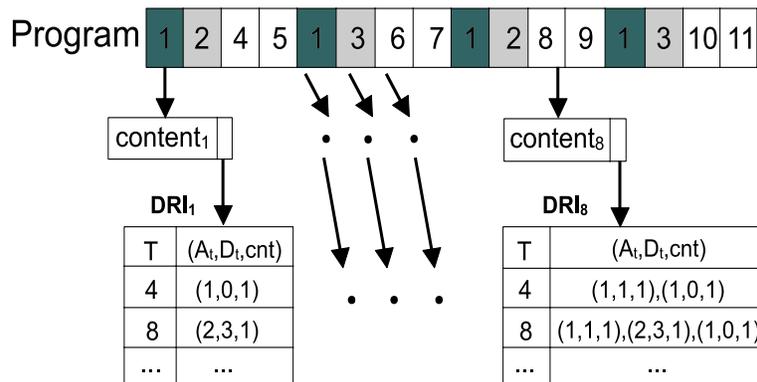


Figure 5: Multiple Data Receiver Indicators

4. Formulation of the problem

From the learner's view point, the greatest benefit is to acquire the most valuable data from a wireless data broadcast system with the least amount of time, no matter at what time instant the learner starts to listen and the total learning time. Hence, we formally define our problem as follows.

Problem formulation:

Given a frequency-based broadcast program B , MU's total turnaround time T and start time S_i .

Find the $(A_i, D_i)^{opt}$ pairs such that the *benefit* is maximal

In the formulated the problem, we observed two phenomenons. First, we observed the watch time affecting the *benefit*. Basically, the less access time has the larger the *benefit*, because spending less time on watching course content means more energy is saved for future use. Second, we also observed broadcast frequency affects the *benefit*. This is because that the higher broadcast frequency represents the more popular course, and the more popular course contains the most applicable data (e.g., hot data) to satisfy learners. As mentioned above, two observations are adopted to express the *benefit*.

- **Observation 1:** The greater the total rest time, the greater the benefit, i.e., *benefit* is proportional to total rest time. This is because the user spends less access time on accessing the needed data.
- **Observation 2:** The greater the importance of the retrieved distinct data, the greater the *benefit*, i.e., *benefit* is proportional to the sum of the broadcast frequency of retrieved distinct data.

Notice, we only consider the distinct data in Observation 2 because a received duplicate data does not offer any new information. Hence, the importance of a retrieved course can be properly represented by its access frequency. These two observations will be applied in our design of the algorithms to enhance efficiency. What is most advantageous for the users is using the least amount of power to acquire the most valuable data. That means the users must make the best of their limited power by maximizing the efficiency of every unit of power. Therefore, we formally define the assessment formula of the *benefit* as

$$benefit = (\text{sum of the frequency of accessed distinct data}) / \text{total consumed power}$$

The higher the broadcast frequency of distinctive data in a given unit of time, the more effectively a user can utilize the power to retrieve data. The broadcast frequency of data in a given unit of time is equal to the importance of the data retrieved in a given unit of time; the higher the importance is, the higher the benefit is. Therefore, our research is simplified to the search for the optimal access behavior of the user that maximizes the broadcast frequency of access in a given unit of time.

5. Benefit-Oriented Data Retrieval Algorithms

A mobile user is unable to know the content on a broadcast server and the broadcast method in

advance, and the mobile user may tune in to the broadcast channel to monitor data at different times. In order to allow mobile users not to waste power on retrieving undesired or unimportant data, the server should provide related information upon mobile users' tuning in to the broadcast channel or at the appropriate time, guiding mobile users to retrieve desired data with an effective method. In a wireless broadcast environment the most frequently used power-saving method for mobile devices is adding an index to broadcast data, using the index to tell the user when the desired data will be broadcast. When the data being broadcast is not desired by the mobile users, doze mode can be selected and awakened to active mode to retrieve data when the desired data is being broadcast. Retrieving data with selective tuning can save a lot of power.

However, the index approach must first retrieve the index packet before knowing when the desired data will be broadcast. The time spent on waiting for the retrieve of the index packet is called initial probe waiting time. The length of such time influences the consumption of power. The influence on initial probe waiting time is related to the amount of index data in the program. Therefore, indexing is divided into single-index approach and multi-index approach according to the amount of index data added to the broadcast data. In this paper, we propose a new method of adding an access mode indicator, or DRI, to the broadcast data. DRI is also categorized as single-mode DRI approach and multi-mode DRI approach according to whether the provided access mode is fixed single set or changeable access mode. Below is the detailed description of our approaches.

5.1 Single-mode DRI Approach

Index technology is an improvement on traditional broadcast index and is quite easy to implement. However, it also inherits the disadvantages of traditional broadcast index. The major shortfalls of index technology include: (1) power wastage while waiting for index packets, and (2) index technology works better for retrieving known data and supports random access environment; if a user does not know the content on the broadcast server in advance, then its efficiency will decrease dramatically.

These two shortfalls are further elaborated here. First, if the data being broadcast when a mobile user tunes in at the time of S_t is not the index packet, then the mobile device goes into doze mode until the index packet is broadcast. This period of time is called initial probe waiting time, during which no data is retrieved and power is wasted. For single-index approach, the initial probe waiting time is about half the time of a broadcast cycle, and the power wasted is broadcast cycle divided by 2 and then multiplied by the power used in doze mode. The second shortfall is that the index approach must begin retrieving data from the most frequently broadcast data until it has been completely retrieved then begin retrieving the less frequently broadcast data until the turnaround time T ends in order to reduce the number of times in active mode. Therefore, the data adjacent to the most frequently broadcast data that is relatively less frequently broadcast is ignored temporarily. However, despite the fact that retrieving the more frequently broadcast data may reduce the tuning time in active mode, it does not necessarily reduce total power consumption because there will still be a long time before the next most frequently broadcast data page is broadcast. This causes a mobile device to remain in doze mode for a long time, waiting for the next most frequently broadcast data to appear. More time is wasted in doze mode and tuning time is increased. The main reason is that the index approach only considers minimizing the number of times in active mode for mobile devices without considering overall optimization.

With the brand new DRI approach that we propose, mobile users may tune in at any t , and begin data retrieving and querying DRI table according to the user's requirement of turnaround time T ; by following the access behavior guided by DRI table, the mobile user may realize the maximum benefit without having index approach related problems. To find the way that allows mobile devices to retrieve the largest possible amount of important data within a limited time T with the

least possible power consumption to achieve maximum benefit, the most intuitive method is brute forced exhaust search (ES). It tests all possible access behaviors and finds the one that gives the greatest benefit. Of course this approach comes with an enormous cost of computation. To reduce such cost of computation, Heuristic Algorithms proposed in our previous work [7] performs very well. Heuristic Algorithms not only reduce the cost of computation, but also achieve near-optimal result. In order to emphasize the necessity of Multi-mode DRI Approach, instead of making comparisons among other Heuristic Algorithms for single-mode DRI approach, this work compares the results of the experiments directly with the ES method that gives the best possible result. A detailed description of the ES method can be found in [7].

5.2 Multi-mode DRI Approach

In Section 2 we have mentioned that the experimental environment that we adopt is broadcast disk because it broadcasts high access frequency data for multiple times to reduce average access time. However, repetitive data may be retrieved in later time if data retrieving behavior is not properly arranged. In single-mode DRI approach, access mode is fixed and some time is wasted under active mode in retrieving repetitive data after several accesses. This not only wastes power on a mobile device, the repetitive data brings no new information and does not help mobile users. Therefore, for mobile users, a good access behavior should be that every data retrieved in active mode can bring new information without repetitive data. Therefore, mobile users should adjust their access behavior at the proper setting/location to maintain the efficiency of the access mode by avoiding adjustment/places where access efficiency reduces progressively or repetitive data may be retrieved. For multi-mode DRI approach, the intuitive method is testing all possible combinations of access behavior to find a list of access behavior pairs, allowing mobile users to retrieve desired data with the least possible power consumption and maximize the benefit. However, finding the best access behavior pairs with exhaust search shall entail a much higher cost of computation when compared with single-mode DRI approach because finding the best access behavior pairs with exhaust search is much more complex than with single-mode DRI approach. For multiple pairs of best access behavior, it is impossible to know in advance if it is best to change the combination of access behavior pairs for twice, or three times, or even more times. Therefore, it is necessary to test all possible combinations of access behavior pairs.

However, it is impossible to know in advance the access mode of every pair of (A_i, D_i) among all possible combinations of access behavior pairs. The cost of computation is far greater than single-mode DRI approach. For multi-mode DRI approach, exhaust search method is not viable, not even randomly selecting a series of access modes. For single-mode DRI approach, random selection by access mode combination method selects one pair of (A_i, D_i) and repeats such action over and over until it reaches the number of experiments that is statistically meaningful [20]. But, multi-mode DRI approach has more than one access mode; it is a serial access mode combination. Therefore, for a longer A_i, D_i , its scope of selection is divided into several parts. The effective scope of randomly selected A_i, D_i from a different access mode combination will be different, too, making random selection method unrealistic. Therefore, it is extremely difficult to find the optimal multiple random selections.

However, in our research, we have designed a simple but effective algorithm based on the two observed rules. We divide multi-mode DRI approach into Greedy Phase and Refinement Phase. Our goal is to find the best benefit. The rule is easy: when a mobile user retrieves this data page, a new benefit is generated; this new benefit is larger than the benefit of the data page that is not yet retrieved. Therefore, retrieving this data page will obviously bring more benefit to the mobile user, so that the mobile user should retrieve this data page. In order to allow this entire combination of access mode to achieve the best benefit, our approach is to first retrieve the data that gives the most benefit or the data with the highest access frequency. Therefore, we rank data pages according to their access frequency for retrieving from high to low (in fact, the broadcast disks

have already been arranged: disk1>disk2> disk3).

From the illustration of Figure 1 we can see that the sequence of retrieve begins with disk1. After all pages on disk1 have been retrieved, disk2 follows, then disk3. If no benefit is added in the process then the retrieve ends. Therefore, the optimal retrieve behavior depends on how many data pages should be retrieved from each disk (disk1, disk2, and disk3). With given T, we will know how many data pages should be retrieved from each of the disks to achieve the optimal combination of data retrieve, and eventually, the maximized benefit for the mobile users. Expressed in the formula for DiskAccessMode(num₁, num₂, ..., num_k), num_i indicates the number of data pages to be retrieved from disk_i. DiskAccessMode is the number of data pages that the mobile user should retrieve from different broadcast disks to satisfy the user's needs and achieve the maximized benefit. With the mobile user's S_t, we can find out which access behavior that the mobile user must adopt to achieve DiskAccessMode. With such access behavior, power consumption can be calculated and a near solution will be available. This is our Greedy Phase. Noticeably, because of the constraint of the mobile user's turnaround time, it is necessary to find out the number of data pages that can be retrieved from each of the disks within the given turnaround time.

Data retrieve in Greedy Phase begins with the high access frequency data, and therefore the desired data can be retrieved with the least possible times in active mode. However, Greedy Phase can only guarantee the least possible times in active mode; it can't guarantee the best power saving. In the process of retrieving high access frequency data, the data page retrieved previously and the data page to be retrieved next time may have a great distance in between; that means retrieving the next data means going through a long period of doze mode, which also consumes power, causing more wastage of power. Perhaps it may do more good by using the power that is to be wasted on retrieving of the next highest access frequency data to increase the total frequency of data retrieve. Despite more time spent in active mode, it greatly reduces the power consumption in doze mode and reduces the total power consumption. Another important benefit is waiting time is also reduced; this is quite different than the previous approaches in which data retrieve continues until turnaround time T is ended. Mobile users may retrieve the next highest access frequency data pages nearby without spending a long period of time waiting for the retrieve of the highest access frequency data, therefore the data retrieve can be completed earlier. Mobile users can proceed to the next task without having to wait for the end of turnaround time T. Data can be fed to users faster, and the mobile users may use the saved time on other work. This is our refinement phase.

6. Performance Study

6.1. Experimental Settings

The mobile system has two components: the *broadcast server* and the *mobile client* (MC). The modeling of the operations in these two components is detailed in the following.

Broadcast Server:

Down at the bottom of the broadcast server in the figure is a *database*, which contains all the data to be broadcast. These data pages will be used to generate the broadcast program in our experiments. In the broadcast program generator, two types of broadcast programs are generated, *Short* and *Long*. In a *Short Program* (denoted as *SP*), the broadcast frequencies of the broadcast disks are of a simple ration $x^n: x^{n-1}: \dots : x: 1$ [17, 18]. Data pages that are not broadcast in such a simple ratio will result in a longer broadcast program, named *Long Program* (denoted as *LP*). In

this performance study, the length of a *LP* is about twice the length of a *SP*. Using both types of broadcast programs in the experiments makes our experiment results more general and more representative. A generated broadcast program is sent to the next module, the index or DRI generator. According to different index or DRI approaches, index data is added to the broadcast programs as an index information packet, and RDI table which consists of different turnaround times is inserted to each data page in the broadcast program. According to the different approaches of index and DRI, there can be two methods of adding index information and DRI table: single-index and multi-index for index and single-mode and multi-mode for DRI Broadcast program with inserted index information or DRI table are then sent to the public channel for mobile users' use.

Table 1 shows the parameters for broadcast server. It explains the parameters used in our performance study and their meanings. The determination of the parameters is based on important literature on broadcast disk technology [17]. NumDisks means the number of broadcast disks on the broadcast server; it is set to 5. DiskSize_{*i*} means the number of data pages on every disk. Whether SP or LP, for convenient comparison on their difference, the numbers of data pages are all set to 1600. The number of data pages contained in every disk is listed in Table 1. The basic principle is: the higher the access frequency, the fewer data pages on a disk; the lower the access frequency, the more data pages on a disk. The calculation of the number of data pages on each disk is based on DiskFreq_{*i*} on each disk and is generated according to the program that generates the broadcast disk program. The readers may refer to [17] for detailed algorithm.

Table 1: Parameters and settings of the broadcast server.

Parameter	Description	Setting
<i>NumDisks</i>	The number of disks in broadcast program.	5
<i>DiskSize_i</i>	The size of disk <i>i</i> , $1 \leq i \leq NumDisks$.	<i>SP</i> : 10 , 40 , 140 , 360 , 1040 <i>LP</i> : 35 , 90 , 315 , 420 , 730
<i>DiskFreq_i</i>	The broadcast frequency of disk <i>i</i> , $1 \leq i \leq NumDisks$	<i>SP</i> : 16 : 8 : 4 : 2 : 1 <i>LP</i> : 9 : 7 : 5 : 3 : 1

Mobile Client:

The *broadcast receiver* in Figure 6 is a component designed to retrieve broadcast pages from wireless channel. Then the downloaded pages are saved and managed in the *client cache component*. The *start time generator* is to generate the start time (*S_i*) and turnaround time (*T*) in the simulation process. Table 2 lists the parameters and their default values used for a MU. The turnaround time *T* varies widely in two types of programs. This is to meet the MU's need as much as possible for a MU could come with various permitted *T*'s. The start time *t_s* can be at any time instance in a major cycle, which means there will be two many choices and that will simply complicate the simulation process. To simply the work, a MU is set to start at only one of the three entering points: *hot*, *medium*, *cold*, meaning that a MU only starts retrieving pages on broadcast disk 1, 3, or 5. To achieve statistically acceptable experimental data, we have simulated the data 30 times to obtain the average values for the results of our experiment. Generally

speaking, in order to save power, typical mobile devices support two operation modes: active mode and doze mode. A typical wireless PC card consumes 60mW in doze mode and 805-1400mW in active mode [13].

In order to reduce the complexity of the experiment, we chose to ignore other power-consuming parts in the query processing with the premise; that the result of the experiment is not affected. Assuming the total power consumption for retrieving one single data page is 1200mW, the formula for calculating total power consumption is $P = 1200 \times Time_{active} + 60 \times Time_{doze}$. In the performance comparison we have considered the optimal access mode that gives the best benefit and least waiting time to allow users to obtain data faster and to use the saved waiting time for other tasks.

Table 2: Parameters and settings of an MH.

Parameter	Settings
T	SP : vary from 200 to 2800 LP : vary from 200 to 2800
S_i	(hot , medium , cold)
<i>Active mode</i>	1200mW
<i>Doze mode</i>	60mW

6.2. Categories of Experiments

In order to make complete comparisons, we categorize our experiments into six categories based on two major factors, the broadcast structure (determined by the server) and the start time of MH, i.e., t_s (determined by the client). The broadcast program structure includes SP and LP , and the start time could be hot, medium, or cold. These six scenarios represent the scenarios that the MH is listening to various broadcast programs at varying start time. They cover all major combinations of the types of clients and servers.

6.3. Performance Results

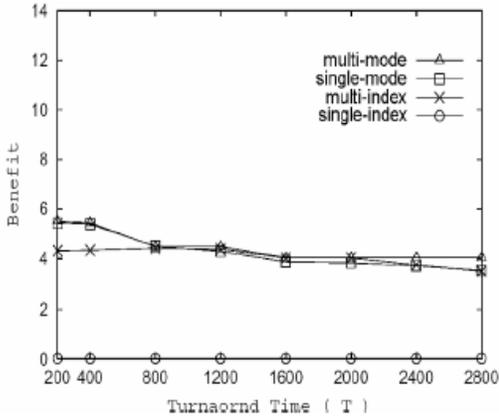
6.3.1 Benefit of the Proposed Algorithms

This experiment is to measure the benefit of the proposed algorithms for different T 's in the six scenarios. The results are given in Figure 8. The X-axis is the turnaround time (i.e., T), and the Y-axis is the *benefit*. The results of the experiment suggest that regardless of the tune-in time of mobile users and broadcast program, multi-mode DRI approach gives the best benefit, followed by multi-index, and single-index has the least benefit. This matches with our analysis above.

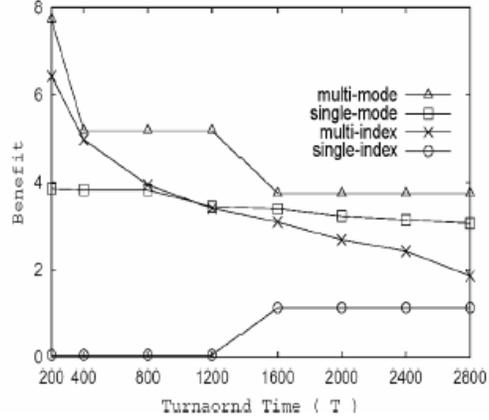
In addition, benefit shall decline as turnaround time grows longer. Data retrieve begins with high access frequency data; the lower the access frequency of the data that is retrieved in later time, the less the benefit is. This explains that the function of benefit declines progressively along with turnaround time. This matches with the progressively decreased marginal efficiency in the supply-demand theory.

On top of the experiment results above, the observation of the experiment also suggests that DRI, as we proposed, also suffers from the phenomenon that the benefit will decrease progressively as turnaround time T grows longer; however, the turnaround T curves in short program 1200 and

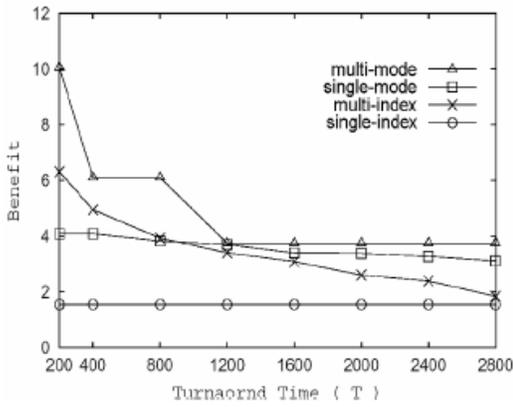
long program 1200 begin to maintain level and stop dropping. It suggests that regardless of the length of mobile users' turnaround time, the system can take the initiative and inform the users that, in SP and LP, data retrieve will only proceed until the turnaround times equal 1200 and 1600, and the users may use the remaining time for other tasks.



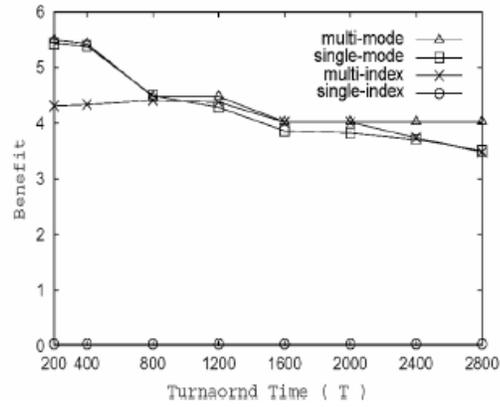
(a) Scenario 1 (SP; $St = hot$)



(b) Scenario 2 (SP; $St = medium$)



(c) Scenario 3 (SP; $St = cold$)



(d) Scenario 4 (LP; $St = hot$)

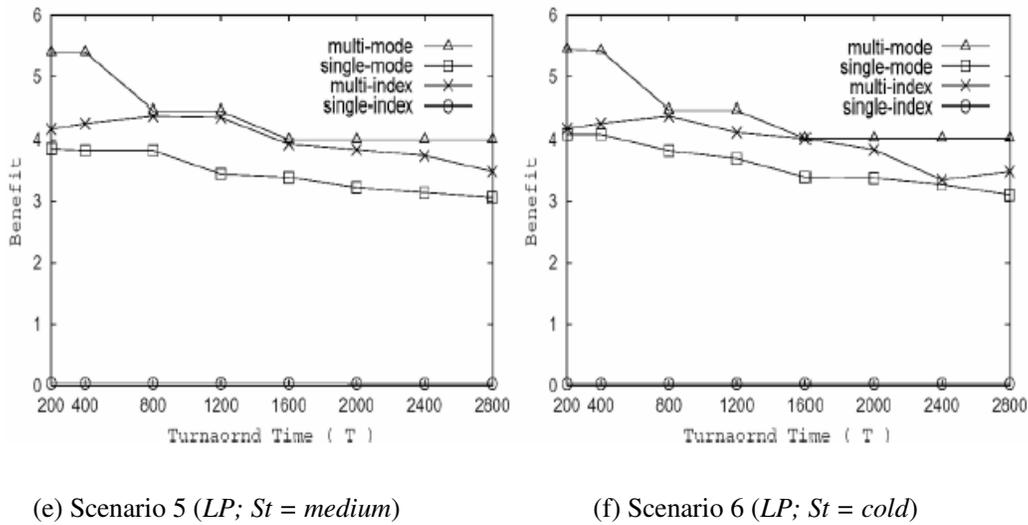
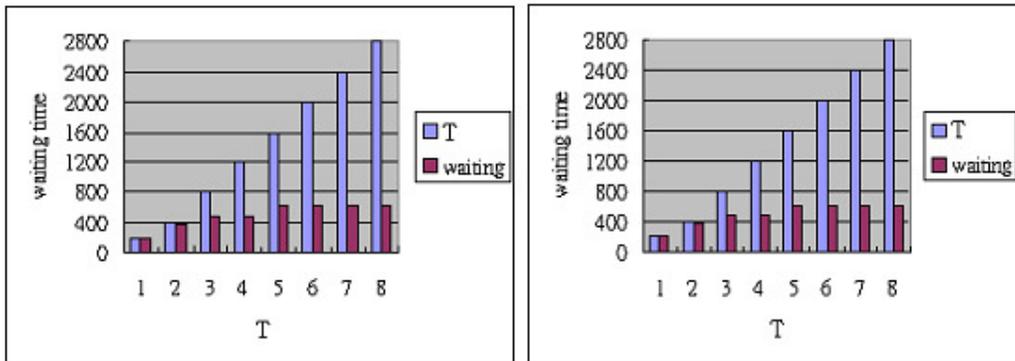


Figure 8: benefits of six scenarios.

6.3.2 Waiting Time in Multi-mode DRI Approach

In addition to the results above showing that the multi-mode DRI approach we propose gives the best benefit, multi-mode approach has another important feature not shared by other approaches: waiting time in multi-mode DRI approach does not have to be equal to turnaround time T , and the waiting time can be shorter. Not only may users receive faster response to their queries, users can also use the saved time for other tasks. The results of the experiment are shown in Figure 9. The X-axis is turnaround time that begins from 200 (represented by 1) to 2800 (represented by 8); a total of 8 classes. T in the marked data on the far right of the Figure is the user's turnaround time. Waiting means the time that must be spent on waiting to achieve the best benefit with a given turnaround time T . The result suggests that the larger T is the more waiting time is saved. Regardless of which type of broadcast program is involved and when the user tunes in, the waiting time can be saved by as much as 75%. This result further confirms the superiority of multi-mode DRI approach.



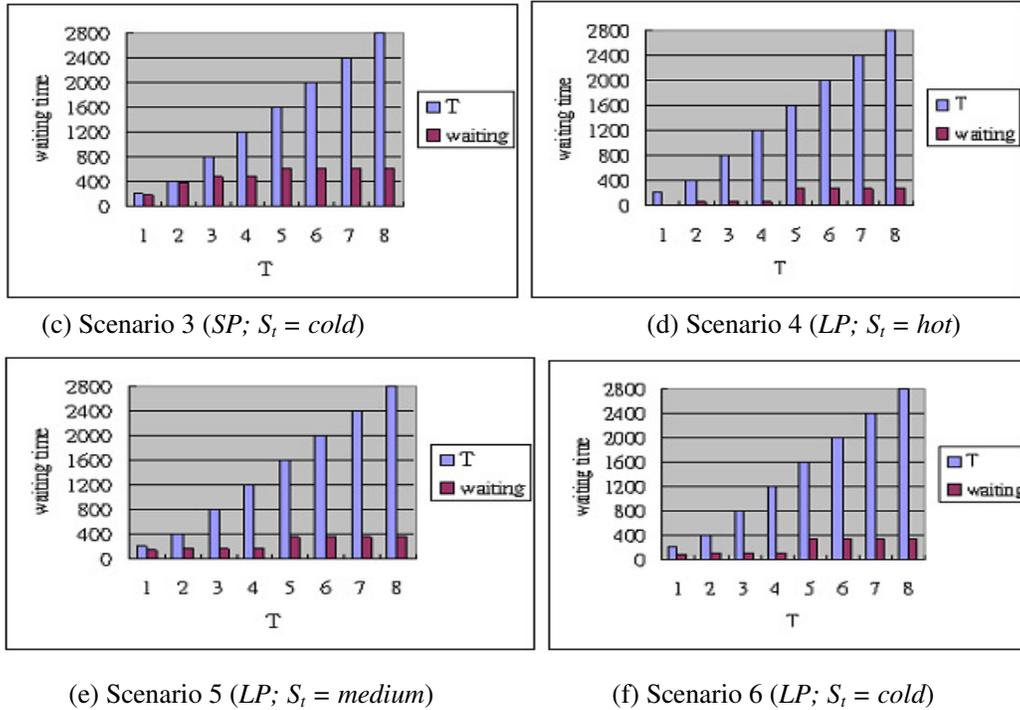


Figure 9: waiting time of six scenarios.

7. Conclusions and Future Work

In this work, we have discussed how to arrange the broadcast schedule to allow the broadcast information system to actively provide enough access modes for mobile users. Mobile users only need to follow this access mode and they may retrieve sufficient data with the best power consumption. The DRI approach that we propose adds DRI table to broadcast data. By doing so, it does not matter when the mobile user tunes in to the broadcast channel, with the information provided in the DRI table they will be able to find the access mode that best fits their needs of query immediately. The DRI approach proposed by us is far better than other improved broadcast index approaches designed to solve the similar problem. Currently, we are studying process single channel broadcast; in the future, we are applying the concepts to data retrieval in multiple broadcast channels.

Acknowledgement

This work was supported by National Science Council of Taiwan (R.O.C.) under Grants NSC100-2221-E-244-018.

REFERENCES

- [1] A. Datta, D.E. Vandermeer, A. Celik, and V. Kumar, (1999) "Broadcast Protocols to Support Efficient Retrieval from Database by Mobile Users", ACM Transactions on Database Systems, Vo1. 24, No. 1, pp. 1-79.
- [2] AirMedia Live Internet Broadcast Networks, <http://www.airmedia.com/> (2010).
- [3] B. Zheng, and D. L. Lee (2005), "Information Dissemination via Wireless Broadcast", Communication of ACM, 48(5), pp. 105-110.
- [4] B. Zheng, J. Xu ,W. C. Lee and D. L. Lee (2004), "Energy Conserving Air Indexes for Nearest

Neighbor Search”, Proceedings of the 9th International Conference on Extending Database Technology, pp. 48-66.

- [5] Cell Broadcast Forum, “<http://www.cellbroadcastforum.org/>” (2010).
- [6] C. C. Chen, C. Lee, and L. F. Lin (2008), “A Novel Encoding Scheme and Data Delivery in Ubiquitous Computing Systems”, *Wireless Networks(WINET)*, 14(6), pp. 731-743.
- [7] L. F. Lin, C. C. Chen, and C. Lee (2010), “Benefit-Oriented Data Retrieval in Data Broadcast Environment”, *Journal of Wireless Networks(WINET)*, 16(1), pp. 1-16..
- [8] E. Yzjima, T. Hara, M. Tsukamoto, and S. Nishio (2002), “Scheduling and Caching Strategies for Broadcasting Corrected Data”, in *Proceedings of the ACM Symposium on Applied Computing*, pp. 504-509.
- [9] E.Pitoura. and P. Chryanthi (2002), “Multiversion Data Broadcast”, *IEEE Transactions on Computer*, Vol. 51, No. 10, pp. 1224-1230.
- [10] I. R. Chen, N. A. P., and I. L. Yen (2002), “Algorithms for Supporting Disconnected Write Operations for Wireless Web Access in Mobile Client-Server Environments”, *IEEE Transactions on Mobile Computing*, Vol. 1, No. 1, pp. 46-58.
- [11] J. H. Hwang, S. Cho, and C. S (2001), Hwang, “Optimized Scheduling on Broadcast Disks”, in *Proceedings of the International Conference on Mobile Data Management (MDM)*, p.p. 91-104.
- [12] Microsoft Corporation. What is directband network? “<http://www.microsoft.com/spot/direct.mspx>”, (2003).
- [13] M. A. Viredaz, L. S. Brakmo, and W. R. Hamburg (2003), “Energy management on handheld devices”, *ACM Queue*, 1(7), pp. 44-52.
- [14] M. Weiser, "Hot Topics: Ubiquitous Computing", *IEEE Computer* (1993).
- [15] Q. Hu, W. C. Lee, and D. L. Lee (1999), “Indexing Techniques for Wireless Data Broadcast under Data Clustering and Scheduling”, in *Proceedings of the International Conference on Information and Knowledge Management (CIKM)*, pp. 351-358.
- [16] R.H. Katz (1994), "Adaption and Mobility in Wireless Information Systems", *IEEE Personal Communications*, Vol. 1, No. 1, pp. 6-17.
- [17] S. Acharya, R. Alonso, M. Franklin, and S. Zdonik (1995), “Broadcast Disks: Data Management for Asymmetric Communications Environments”, in *Proceedings of the ACM SIGMOD Conference*, pp.199-210.
- [18] S. Acharya, M. Franklin, and S. Zdonik (1995), “Dissemination-Based Data Delivery Using Broadcast Disks”, *IEEE Personal Communications*, 2(6), pp. 50-60.
- [19] S. Acharya, M. Franklin, and S. Zdonik, (1996), “Prefetching from a Broadcast Disk”, in *Proceedings of the International Conference on Data Engineering*, pp. 276-285.
- [20] S. M. Ross (2000), “Introduction to Probability and Statistics for Engineers and Scientists”, John Wiley & Son Inc.
- [21] S. Shekhar ,A. Fetterer and D. R. Liu, Genesis (1996), “An Approach to Data Dissemination in Advanced Traveler Information Systems”, *Data Engineering Bulletin*, Vol. 19, No. 3, pp. 40-47.
- [22] T. Imielinski, S. Viswanathan, and B. R. Badrinath (1997), “Data on Air: Organization and Access”, *IEEE Transactions on Knowledge and Data Engineering*, 9(3), pp.353-372.