Broadcast Scheduling for Multiple Channels in Wireless Information Systems

Chih-Horng Ke
Department of Computer Science and Information Engineering
National Cheng-Kung University
Tainan, Taiwan, R.O.C.
kech@csie.ncku.edu.tw

Abstract

Some characteristics of wireless information systems, such as limited bandwidth and power supply, provide many new challenges for traditional techniques. Data broadcast is considered an efficient way, in terms of energy consumption and bandwidth utilization, for disseminating data to a massive number of mobile clients in such a system. Several papers have been devoted to the issue of broadcast scheduling. Most of them assume that each mobile client needs only one data page. However, in many situations an mobile client might need data of more than one page. Under these circumstances, how to minimize the access time for the mobile clients is a non-trivial task. The scheduling strategy should take the affinity between data pages into consideration. Moreover, the past work tackled the problem mainly from the view point of broadcasting the data pages over single channel. As a number of wireless channels are available, the issue of broadcast scheduling becomes more complicated. In this paper we propose five scheduling strategies for data broadcast.

Keywords: Wireless information system, Broadcast schedule, Multiple wireless channels.

1 Introduction

In a Wireless Information System (WIS), users equipped with mobile computers (MCs) access data through wireless channels at anytime and anywhere. Some characteristics of WIS, such as limited bandwidth of wireless channels and scarce power supply of MCs, provide many new challenges for traditional techniques of a wired information system. In current wired information system, MCs initiate data delivery by sending requests to a server. Then the MCs *pull* data from the server in order to provide data to locally running applications. We refer to such an approach as the pull-based approach. By using this scheme, in a WIS the server will become bottleneck when it services a great deal of MCs. New data delivery mechanisms should be proposed.

To avoid multiple transmissions of commonly requested data, data broadcast, explored by several recent papers [AAFZ95, AFZ95, CK99, IV94, IVB97, TYE99], has been proposed to broadcast popular data such as stock trading, weather, traffic control, etc, on the wireless channels. When an MC needs data pages, it monitors the wireless channels until the desired data pages are detected and captured for use. Because the cost of broadcasting data over the wireless channels does not depend on the number of listening MCs, an unlimited number of MCs can tune into the wireless channels and extract their desired data. And accessing broadcast data does not consume energy for submitting any query. Thus data broadcast is considered an efficient way in terms of energy consumption and bandwidth utilization for disseminating data to a massive number of MCs in a wireless information system.

Over the last few years, a great deal of effort [AAFZ95, AFZ95, CK99, KL96] has been made to minimize the average access time of the broadcast data. [AAFZ95, AFZ95] proposed the mechanism of broadcast disk to provide data access in a WIS. The broadcast disk technique treats a broadcast stream of data that are repeatedly and cyclically transmitted as a storage device. Other works focused on designing different scheduling strategies for the data broadcast. Most of these works assume that each MC needs only one data page and leaves the system when the desired page is obtained. Thus, minimizing the access time of one data page is their main concern of performance criterion. Two exceptions of these researches are the work in [CK99, KL96] where an MC may access more than one data page. In fact, the assumption that each MC needs only one data page is over-simplified. In many situation, an MC might need data of more than one data page. For instance, (1)*Stock market application* : Trading stocks is one of popular investments. Data is periodically broadcast on the wireless channels. Users often need prices of one or more stocks. The desired data by a particular user, although it may not be big, might be scattered in different data pages. (2)*Traffic application* : In the metropolis there are heavy traffic sometime and/or somewhere daily. Users may want to get the traffic information of a few roads, which may again be spread over more than one data page in the broadcast data. (3)*Activity announcement* : Many activities can be held simultaneously. A user may wish to first attend a concert and then go to a computer show. The information about the location and time of these activities are often not in the same pages. To minimize the access time for the MCs under these circumstances is a non-trivial task. As improving the time for accessing a data page might delay the time for accessing other data pages. Moreover, all of the past works discuss how to broadcast the data over *single* wireless channel. [CK99, KL96] discussed how to schedule the broadcast data over single channel. No work focuses on the issue of scheduling the broadcast data on *multiple* wireless channels. In this paper, we study the issue of scheduling the broadcast data on multiple wireless channels as an MC may access more than one data page.

1.1 Motivation

We reveal that the issue of broadcasting data on multiple wireless channels cannot be simply considered as multiple subissues, each broadcasting data on single wireless channel. Let us illustrate this with the following example.

<Example 1>

Five users' requests and their desired data pages are shown in the table. S_i and P_i represent user's request and the data page, respectively, where i is a positive integer. The request ratio of data set S_1 is 30% indicating that on average S_1 are requested three times.in every ten requests.

User's request	Desired data pages	Request ratio
S_1	P_1, P_2, P_4, P_7	30%
S_2	P_4, P_5	25%
S_3	P_3, P_4	20%
S_4	P_1, P_3, P_7	15%
S_5	P_6, P_8	10%

We assume two wireless channels, C_1 and C_2, are available. Time on the wireless channels is divided into slots of same size that is equal to the time to broadcast a data page. Time slot i, named T_i, is the interval [i-1, i). Two different broadcast schedules, M_1 and M_2, are shown as follows. In both M_1 and M_2, the server broadcasts all data pages in each broadcast cycle and two broadcast cycles are presented for each schedule. More specifically, T_1 to T_4 indicates one broadcast cycle, and T_5 to T_8 is next broadcast cycle. Here we assume that each data page is broadcast once within each broadcast cycle.

Page	P_4	P_1	P_7	P_3	P_2	P_5	P_6	P_8
R_i(%)	75	45	45	35	30	25	10	10

M_1:	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
C_1	P_4	P_7	P_2	P_6	P_4	P_7	P_2	P_6
C_2	P_1	P_3	P_5	P_8	P_1	P_3	P_5	P_8

M_2:	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
C_1	P_1	P_2	P_4	P_7	P_1	P_2	P_4	P_7
C_2	P_5	P_3	P_6	P_8	P_5	P_3	P_6	P_8

The first schedule M_1 is arranged based on the request ratios of data pages. Here we define the request ratio of data page P_i, named R_i, as the summation of request ratios of all requests that access P_i. Hence M_1 is obtained by scheduling P_4 at the first time slot T_1 of channel C_1, P_1 at T_1 of channel C_2, P_7 at T_2 of channel C_1, ..., and so on. The second schedule M_2 is arranged based on the request ratio of users' requests. First, because the request ratio of S_1 (=30%) is maximal, all data pages requested by S_1 are scheduled on channel C_1. Then the data pages requested by S_2, i.e., P_4 and P_5, are scheduled. Because P_4 has been arranged on T_3 of channel C_1, the server does not schedule P_4 again and only schedules P_5 at the time slot T_1 of channel C_2. Similarly, the data pages requested by S_3, S_4, and S_5 can be scheduled.

For M_1 and M_2, the expected access time of each MC's request is the time for receiving all desired data pages. *Note that at each time slot an MC can only tune into one channel to retrieve one data page.* Hence, if two of desired data pages (e.g. in M_1, P_4 and P_1 of S_1) are broadcast at the same time slot of different channels, MC will receive one data page broadcast during one broadcast cycle (e.g. receive P_4 at T_1) and receive the other data page broadcast during the next broadcast cycle (e.g. receive P_1 at T_5). Thus the expected access time of MC that requests S_1 is five. Similarly, in M_1 the expected access time of S_2, S_3, S_4, and S_5 are three, two, six, and eight, respectively. Therefore for the schedule M_1 the average access time of MCs' requests is 4.35 (= 5*0.3 + 3*0.25 + 2*0.2 + 6*0.15 +8*0.1). Similarly, the average access time for MCs in M_2 can be computed. We see that the performance of M_2 is better than that of M_1.

There are three major points that are demonstrated by this example. The *first* point is that the expected access time of MC's request is determined by the time that the MC receives the last of its desired data pages. Thus the issue of scheduling the broadcast data for the situation that each MC may access multiple data pages cannot be simply considered as multiple subissues, each scheduling the broadcast data for the situation that each MC only retrieves one data page. The *second* point is that the schedule M_2 performs better than the schedule M_1. This fact demonstrates that all data pages accessed by a request should be simultaneously considered where to schedule. Scheduling the broadcast data only based on the request ratio of each data page will perform badly. The *third* point demonstrated by the above example is that the issue of scheduling the broadcast data on multiple channels cannot be considered as multiple subissues, each broadcasting data on single wireless channel. This is because the server should avoid to schedule the desired data pages of an MC's request at the same time slot of these channels. Otherwise, the expected access time will be increased. This phenomenon can be observed by S_1 in the schedule M_1.

1.2 Contributions

The above example exhibits that there are some subtle issues resulting from the broadcast scheduling on multiple wireless channels. In this work, we study how to schedule the broadcast data on multiple wireless channels while users' requests may access more than one data page. We propose five strategies for scheduling the broadcast data. All proposed strategies are designed to minimize the expected access time of MCs.

The rest of the paper is organized as follows. In Section 2, we present the environment under which this research is conducted. Two page-based scheduling strategies are proposed in Section 3. Then in Section 4, we describe three set-based scheduling strategies in detail. Finally, we conclude this paper in section 5 and present our future work of this research.

2 Environment

2.1 The broadcast environment

Basically, a wireless information system (WIS) consists of two sets of entities : *servers* and *mobile clients (MCs)*. In general, servers are powerful stationary machines with database systems, and MCs are portable computing devices, e.g. palmtop or laptop computers, running on AA batteries. MCs may be diskless or only equipped with limited amount of storage space. Each MC will access data items via wireless information channels. The *Scheduler* at the server receives the broadcast data and possibly more information such as access patterns of data. It then computes a "good" schedule for the data broadcasting. A database is divided into data pages, and the smallest logical unit of the broadcast data is a *data page*. The time required to broadcast a data page is referred to as a *time slot*. A *broadcast schedule* specifies when and where each data page is to be transmitted. In this work we consider systems with a number of available wireless channels. If B wireless channels are available, we number these channels from 1 to B. Thus a broadcast schedule can be represented by a *broadcast matrix* $M = [m_{ij}]$, where the value of m_{ij} is the data page broadcast at the j-th time slot of the i-th channel. For instance, in Example 1 the broadcast matrix of the schedule M_1 is as follows. A *broadcast cycle* is a sequence of broadcast data pages, P_{i1}, P_{i2}, ..., P_{ic}, where $i \in \{1, 2, \ldots, B\}$ and c is the number of columns of broadcast matrix. Obviously, the length of a broadcast cycle is the number of columns of broadcast matrix. Therefore, if the server starts to broadcast the data over all channels at time 0, P_{ij} will be broadcast at the j-th time slot, $(j + c)$-th time slot, ..., and so on.

$$\begin{bmatrix} P_4 & P_7 & P_2 & P_6 \\ P_1 & P_3 & P_5 & P_8 \end{bmatrix}$$

2.2 Assumptions and relevant issues

Because MCs are only powered by limited-energy batteries, data access strategies should be designed to minimize the MCs' energy consumption. This is realized by broadcasting the index pages with data pages [IVB97, TO98, TY97]. Besides the advantage of minimizing energy consumption, more importantly the index serves the purpose of allowing the MCs to selectively tune into the channels in order to access the desired data pages when they are broadcast. Therefore, an MC needs to get the index pages first before it starts to retrieve the desired data pages. There are many techniques described in the literature for indexing [IVB97, TO98, TY97]. Because the design of an indexing scheme and the design of a broadcast schedule are orthogonal, we do not consider the issue of indexing in order to focus on the problem of broadcast scheduling. In this study we simply assume that the index is broadcast at the beginning of each broadcast cycle. Avoiding to involve the detail of indexing techniques, the time of accessing index pages is excluded from the access time of MCs. Hence, in this study the access time of an MC is the time from the moment an MC gets the index pages to the point when the MC receives all the desired data pages. Our scheduling strategies are designed to minimize the expected access time of MCs.

Another issue is how to obtain the statistics of which set of data pages is accessed by an MC. It is because obtaining such feedback is contradictory to the nature and motivation of data broadcast technique which assumes no uplink requests from the MCs. In [IV94] a number of possible solutions were described. Different methods depend on how much *up to date* the server wants to be regarding the distribution of MCs' requests. Here are two possible solutions. In the first method, the MCs informs the server just before disconnection then it can piggyback a message indicating the data pages that it accessed in the duration of connecting to the server. In the second method, server periodically drops one or more pages from the set of broadcast data and provides them on-demand for a while, thereby estimating as to how frequently those pages are requested. The detail of all possible methods for gathering the statistics can be seen in [IV94].

Finally, we introduce some notations used in the rest of this work. Let B be the number of available channels for broadcasting data. C_b is the b-th channel, $b \in \{1,2,\ldots,B\}$. n represents the number of broadcast data pages. P_i is one of broadcast data pages, $i \in \{1,2,\ldots,n\}$. And let q be the number of data sets that are accessed by MCs. S_j is a set of data pages that are accessed by an MC, $j \in \{1,2,\ldots,q\}$. R_i, the *request ratio of a data set*, is the ratio of the number of the set being requested to the number of all data sets within a certain amount of time. F_i, the *access frequency of a data page*, is computed by summing up the product of the number of all data sets and the request ratios of data sets that request this page.

3 The Page-based Scheduling Strategy

We discuss two classes of broadcast scheduling : (1) the strategies of the first class are designed based on the access frequencies of data pages and (2) the strategies of the second class are designed based on the request ratios of data sets. The strategies of the first class are discussed in this section and the strategies of the second class in the next section. All scheduling strategies proposed in this work will illustrate Example 2 for explanation.

<Example 2>

Six data sets and their data pages are shown in the table along with the request ratios of data sets. For the sake of convenience, we will ignore the notation "%" of request ratio in the later illustration. Three channels, C_1, C_2, and C_3, are available.

Data set	Data pages	Request ratio
S_1	P_1, P_2, P_3, P_4	35%
S_2	P_3, P_6	25%
S_3	P_4, P_7, P_8	20%
S_4	P_2, P_9, P_{11}, P_{14}	8%
S_5	P_6, P_{10}, P_{12}, P_{14}	7%
S_6	P_5, P_{11}, P_{13}, P_{15}	5%

3.1 The Page-based strategy – Vertical Version (PVV)

The goal of scheduling is to minimize the access time of MCs, thus MCs should retrieve the desired data pages as fast as possible after receiving the index page. Hence, a simple idea is that a data page with maximal access frequency should be broadcast first. This forms the basis of the first page-based strategy, the *Page-based strategy – Vertical Version (PVV)*. Formally, this strategy is presented as follows.

```
PVV Strategy :
  Input : X, a set of data pages ;
  Output : M, a broadcast matrix ;
  begin
     Given n broadcast data pages P_1, P_2, ..., P_n
        with access frequencies F_1, F_2, ..., F_n ;
     Let B be the number of available channels ;
     Let X = {P_1, P_2, ..., P_n} and M = [m_ij] ;
     Sort P_r on their access frequencies and rearrange
        their index such that F_r >= F_k if r <= k;
     Let i = j = 1 ;
     while X != φ do
        For r = 1 to n do
           Select P_r ;
           Set X = X - {P_r} ;
           If r is a multiplier of B, let j = B ;
              otherwise let j be the remainder of
                 dividing r by B ;
           Set m_ij = P_r ;
           If r is a multiplier of B, let i = i + 1 ;
  end
```

In this algorithm, we show that a data page with a higher access frequency is always broadcast earlier than a data page with a lower access frequency. "$-$" is the set difference operation. The output M of this algorithm is a broadcast matrix of data pages. The server will periodically broadcast the data pages to MCs according to this matrix. We explain how this algorithm works by using Example 2 and show the resulting sequence as follows. For sake of convenience, we ignore notation "%" in the access frequencies of data pages.

Data page	P_3	P_4	P_2	P_1	P_6	P_7	P_8	P_{14}	P_{11}	P_9	P_{10}	P_{12}	P_5	P_{13}	P_{15}
Access frequency	60	55	43	35	32	20	20	15	13	8	7	7	5	5	5

$$M = \begin{bmatrix} P_3 & P_1 & P_8 & P_9 & P_5 \\ P_4 & P_6 & P_{14} & P_{10} & P_{13} \\ P_2 & P_7 & P_{11} & P_{12} & P_{15} \end{bmatrix}$$

Data set	S_1	S_2	S_3	S_4	S_5	S_6
Access time	11	2	3	8	9	15

$$\textit{Total access time} = 11*35+2*25+3*20+8*8+9*7+15*5 = 697$$

In this broadcast matrix produced by the PVV strategy, P_3, P_4, and P_2 have the highest access frequency and can be broadcast at the first time slot of three channels, C_1, C_2, and C_3, respectively. Next, the server will arrange P_1, P_6, and P_7 at the second time slot of three channels. Similarly, other data pages are also assigned to the broadcast matrix. The access time of data set S_1 is 11. This is because P_3, P_4, and P_2 are arranged to the first time slot of three channels and an MC accessing S_1 can only retrieve one data page at each time slot. So if the

MC retrieves P_3 at the first time slot of one broadcast cycle, P_4 and P_2 need to be accessed at the first time slot of the next two broadcast cycles, respectively. The length of broadcast cycle is five. Thus the access time of S_1 is 1+2*5=11. Similarly, the access time of other data sets can be computed and shown as above. The total access time is calculated by multiplying the request ratio of each data set times the access time for that data set and summing the results.

3.2 The Page-based strategy – Horizontal Version (PHV)

In the PVV strategy, the server always broadcasts the data page with the highest access frequency at the earliest time slot. However, broadcasting the data pages with closer access frequencies (e.g. P_2, P_3, and P_4 within S_1) at the same time slot could increase the access time of data set (i.e., S_1). In this subsection, another method is proposed to horizontally spread the data pages into the broadcast matrix. This scheduling strategy is called the *Page-based strategy – Horizontal Version (PHV)*. The basic idea is to sort all data pages by their access frequencies in descending order and sequentially arrange them to the different time slots of the same channel. This procedure is proceeded from arranging the first data page to the first time slot of C_1. As all time slots of C_i within one broadcast cycle are filled, the first time slot of C_{i+1} is used for next data page. This process proceeds until all data pages are arranged to the broadcast matrix. We formally present this strategy in the following.

PHV Strategy :

Input : X, a set of data pages ;
Output : M, a broadcast matrix ;
begin
  Given n broadcast data pages $P_1, P_2, \ldots, P_n$ with access frequencies $F_1, F_2, \ldots, F_n$;
  Let B be the number of available channels ;
  Let $X = \{P_1, P_2, \ldots, P_n\}$ and $M = [m_{ij}]$;
  Sort P_r on their access frequencies and rearrange their index such that $F_r \geq F_k$ if $r \leq k$;
  Let $i = j = 1$ and $T = \lceil \frac{n}{B} \rceil$;
  while $X \neq \phi$ do
    For $r = 1$ to n do
      Select P_r ;
      Set $X = X - \{P_r\}$;
      If r is a multiplier of T, let $j = T$;
        otherwise let j be the remainder of dividing r by T ;
      Set $m_{ij} = P_r$;
      If r is a multiplier of T, let $i = i + 1$;
end

In the following, we also explain how this algorithm works by using Example 2. The produced broadcast matrix and total access time by using the PHV strategy are also shown in the following. P_3, P_4, P_2, P_1, and P_6 are with the highest access frequencies and can be broadcast at five time slots of C_1, respectively. Then P_7 is broadcast at the first time slot of C_2. Other data pages are also scheduled on the channels. Hence, we found that the PHV strategy outperforms the PVV strategy. The PHV strategy, however, can be further improved by considering the relationship between data pages. We will consider this factor in next section.

$$M = \begin{bmatrix} P_3 & P_4 & P_2 & P_1 & P_6 \\ P_7 & P_8 & P_{14} & P_{11} & P_9 \\ P_{10} & P_{12} & P_5 & P_{13} & P_{15} \end{bmatrix}$$

Data set	S_1	S_2	S_3	S_4	S_5	S_6
Access time	4	5	7	8	5	9

$$\begin{aligned} Total\ access\ time &= 4*35+5*25+7*20+8*8 \\ &\quad +5*7+9*5 = 549 \end{aligned}$$

4 The Set-based Strategy

In the above section, the proposed scheduling strategies consider the access frequencies of data pages as the single criterion of scheduling and ignore the relationship between data pages. Hereafter, P_i and P_j are within the same data set and broadcast at the same time slot of different channels, we denote that P_i *conflicts with* P_j. The conflicts will increase the access time of MCs retrieving data sets. In this section three set-based scheduling strategies are designed to minimize the access time of MCs by reducing the conflicts between data pages. The basic idea is to arrange the data pages of the same data set to different time slots.

4.1 The Set-based strategy – Horizontal Version (SHV)

Conceptually, the first set-based scheduling strategy horizontally broadcast the data pages over the wireless channels and put the data pages of the same data set as together as possible. The idea of the first set-based strategy, the *Set-based strategy – Horizontal Version (SHV)*, is described as follows. First, the server sorts data sets S_i by their request ratios in descending order, $i \in \{1,2, \ldots q\}$. Then, for each broadcast cycle, all data pages of S_i are sequentially arranged to the contiguous time slots of available channels except the data page has been arranged. All time slots of channel C_i are used before any one time slot of C_j is used, where $i \leq j$. Formally we present this strategy as follows.

SHV Strategy :

Input : S_i, q sets of data pages, $1 \leq i \leq q$;
Output : M, a broadcast matrix ;
begin
  Given q data sets $S_1, S_2, \ldots, S_q$ with request ratios $R_1, R_2, \ldots, R_q$;

```
Let B be the number of available channels ;
Let X = ∪_i S_i = {P_1, P_2, ..., P_n} be all
    broadcast data pages ;
Let M = [m_ij] and f_i = 0, i∈{1,2,...n} ;
Sort S_r on their request ratios and rearrange
    their index such that R_r ≥ R_k if r ≤ k;
Let i = j = 1 and T = ⌈n/B⌉;
For r = 1 to q do
    while S_r ≠ φ do
        Select P_k, where P_k∈S_r ;
        Set S_r = S_r − {P_k} ;
        If f_k = 0, set m_ij = P_k and f_k = 1 ;
        If j is a multiplier of T,
                let i = i + 1 and j = 1 ;
            Otherwise j = j + 1 ;
end
```

In the SHV strategy, the data pages of a data set with higher request ratio are always scheduled earlier than the data pages of a data set with lower request ratio. $\cup$ is the set union operation. The flag f_i is designed to judge whether a data page has been scheduled. $f_i = 0$ represents that P_i has not been scheduled and $f_i = 1$ means that P_i has been scheduled.

$$M = \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_6 \\ P_7 & P_8 & P_9 & P_{11} & P_{14} \\ P_{10} & P_{12} & P_5 & P_{13} & P_{15} \end{bmatrix}$$

Data set	S_1	S_2	S_3	S_4	S_5	S_6
Access time	4	5	4	5	10	9

$$\begin{aligned} \textit{Total access time} &= 4*35+5*25+4*20+ \\ &\quad 5*8+10*7+9*5 = 500 \end{aligned}$$

Now we explain how this algorithm works by using the data given in Example 2. The broadcast matrix generated by using the SHV strategy is shown as above. First, the request ratio of the data set S_1 is largest, four data pages of S_1 (i.e., P_1, P_2, P_3, and P_4) are scheduled at the first four time slots of C_1, respectively. Next, all data pages of S_2 should be scheduled. Because P_3 has been scheduled at the third time slot of C_1, the server needs only to schedule P_6 at the fifth time slot of C_1. Then the server schedules the data pages of S_3. P_7 and P_8 will be scheduled at the first and second time slots of C_2 because all time slots of C_1 within a broadcast cycle have been used and P_4 has been scheduled. Similarly, the data pages of other data sets can be scheduled. Hence we can compute the total access time of MCs generated by using the SHV strategy. Comparing the results with those given in the PVV strategy and the PHV strategy, we see that the SHV strategy performs better than the page-based strategies.

4.2 The Set-based strategy – Vertical Version (SVV)

Similar to the page-based strategies, we can design the set-based strategy by scheduling the data pages of a data set in various directions. In the first set-based strategy, the server attempts to fill all time slots of the first available channel with the un-broadcast data pages before any time slots of other channels are used. Such an action can be viewed as horizontally scheduling the data pages. One of its drawbacks is that some data page of a data set with high request ratio may be scheduled at a later time slot, e.g. P_6 of S_2 is scheduled at the fifth time unit of C_1. Thus, in this subsection we consider to schedule the data pages in another direction. In order to improve the above drawback of the SHV strategy, the second set-based strategy broadcasts the data pages of the data sets with higher request ratios as early as possible. The basic idea of this strategy is that if the data pages of data set S_i are broadcast over channel C_j, the data pages of data set S_{i+1} are broadcast over channel C_{j+1}. This process is like a round-robin procedure. We name the second set-based strategy the *Set-based strategy – Vertical Version (SVV)*. We formally present the SVV strategy as follows.

```
SVV Strategy :
  Input : S_i, q sets of data pages, 1≤ i ≤ q ;
  Output : M, a broadcast matrix ;
  begin
    Given q data sets S_1, S_2, ..., S_q with request
        ratios R_1, R_2, ..., R_q ;
    Let B be the number of available channels ;
    Let X = ∪_i S_i = {P_1, P_2, ..., P_n} be all
        broadcast data pages ;
    Let M = [m_ij] and f_i = 0, i∈{1,2,...n} ;
    Let E_j = 1, j∈{1,2,...B} ;
    Sort S_r on their request ratios and rearrange
        their index such that R_r ≥ R_k if r ≤ k;
    Let T = ⌈n/B⌉ and i = j = 1 ;
    For r = 1 to q do
        while S_r ≠ φ do
            Select P_k, where P_k∈S_r ;
            Set S_r = S_r − {P_k} ;
            If f_k = 0
                while E_i > T do
                    set i = i + 1 ;
                Set j = E_i and E_i = E_i + 1 ;
                Set m_ij = P_k and f_k = 1 ;
        If r is a multiplier of B, let i = 1 ;
            otherwise, i = i + 1 ;
  end
```

In the SVV strategy, the data pages of a data set with high request ratio are broadcast as early as possible. Hence the server arranges all data sets to the available channels in a round-robin pattern. The last two lines of the above algorithm are designed to achieve such a function. The variable E_j is used to record the first free time slot of channel C_j. As all time slots of C_i are used, the server choose next channel that has free time slots. The loop, *while* $E_i > T$, is designed for this intention. Again, we use the data given in Example 2 to illustrate how this algorithm works.

$$M = \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_9 \\ P_6 & P_{11} & P_{14} & P_{10} & P_{12} \\ P_7 & P_8 & P_5 & P_{13} & P_{15} \end{bmatrix}$$

Data set	S_1	S_2	S_3	S_4	S_5	S_6
Access time	4	3	4	7	5	5

$$\textit{Total access time} = 4*35+3*25+4*20+7*8+5*7+5*5=411$$

The broadcast matrix generated by using the SVV strategy is shown as above. First, the request ratio of the data set S_1 is highest, four data pages of S_1 (i.e., P_1, P_2, P_3, and P_4) are scheduled at the first four time slots of C_1, respectively. Next, the data pages of S_2 should be scheduled on channel C_2. Because P_3 has been scheduled at the third time slot of C_1, the server needs only to schedule P_6 at the first time slot of C_2. Then the server schedules the data pages of S_3 on channel C_3. So P_7 and P_8 are scheduled at the first and second time slots of C_3, respectively. P_4 does not need to be scheduled because it has been arranged at the fourth time slot of C_1. In this example, we assume that three wireless channels are available for broadcasting data pages. Hence, the data pages of S_4 should be scheduled on channel C_1. However, P_{11} and P_{14} are scheduled at the second and the third time slots of C_2, respectively. This is because C_1 has no free time slot after scheduling P_9 at the fifth time slot of C_1. Similarly, the data pages of other data sets can be scheduled over wireless channels. Then the total access time of using the SVV strategy can be computed. It indicates that the SVV strategy outperforms the preceding three scheduling strategies.

4.3 The Set-based strategy – Non-overlap Version (SNV)

The above four scheduling strategies cannot reduce the probability of conflicts between data pages to zero. There is still some room for further reducing the probability that the conflicts appear. We use the data given in Example 2 as an example for illustration. In the broadcast matrix generated by using the SVV strategy, P_2 and P_{11} are broadcast at the second time slot of C_1 and C_2, respectively. Hence an MC accessing S_4 must receive P_2 (or P_{11}) during one broadcast cycle and P_{11} (or P_2) during the next broadcast cycle. So the access time of retrieving data set S_4 is seven. In this subsection, the third set-based strategy is considered to avoid such a phenomenon and further reduce the conflicts between data pages. The basic idea of this strategy, named the *Set-based strategy – Non-overlap Version (SNV)*, is similar to that of the SVV strategy. Conceptually, the SNV strategy also arranges all data sets to the available channels in a round-robin pattern. However, the SNV strategy also tries to avoid scheduling two or more data pages of one data set at the same time slot of different channels. Formally, this strategy is presented as follows.

SNV Strategy :

Input : ξ_i, q sets of data pages, $1 \leq i \leq q$;
Output : M, a broadcast matrix ;
begin
  Given q data sets $\xi_1, \xi_2, \ldots, \xi_q$ with request ratios $R_1, R_2, \ldots, R_q$;
  Let B be the number of available channels ;
  Let $X = \cup_i S_i = \{P_1, P_2, \ldots, P_n\}$ be all broadcast data pages ;
  Let $S_i = \xi_i$, $A_i = \phi$, and $T = \lceil \frac{n}{B} \rceil$;
  Let $M = [m_{ij}]$ and $f_i = 0$, $i \in \{1,2,\ldots n\}$;
  Let $E_{ij} = j$, $i \in \{1,2,\ldots B\}$, $j \in \{1,2,\ldots n\}$;
  Sort S_r on their request ratios and rearrange their index such that $R_r \geq R_k$ if $r \leq k$;
  Let $i = j = 1$;
  For $r = 1$ to q do
    While $S_r \neq \phi$ do
      Select P_k, where $P_k \in S_r$;
      Set $S_r = S_r - \{P_k\}$;
      while $f_k = 0$
        while $E_{ij} > T$ and $A_i = \phi$ do
          set $i = i + 1$;
        If $\exists\ E_{iu} \in A_i$ s.t. no P_h is in S_r and satisfies $m_{xu} = P_h$ for some x ;
          Set $m_{iu} = P_k$ and $f_k = 1$;
          Set $A_i = A_i - E_{iu}$;
        If $E_{ij} \leq T$ and $f_k = 0$
          while $\exists\ P_h \in \xi_i$ and $m_{xj} = P_h$ for some x ;
            Set $A_i = A_i \cup \{E_{ij}\}$;
            Set $j = j + 1$;
          If $j > T$
            Set $i = i + 1$;
          else
            Set $m_{ij} = P_k$ and $f_k = 1$;
            Set $j = j + 1$;
      If $f_k = 0$ and $E_{ij} > T$ for all i
        Set $m_{Bj} = P_k$ and $f_k = 1$;
        Set $j = j + 1$;
    If r is a multiplier of B, let $i = 1$; otherwise, $i = i + 1$;
end

In the SNV strategy, E_{ij} represents the j-th time slot of C_i. A_i is designed to store the time slots of channel C_i that causes the conflicts between some data pages of an data set. More precisely, if $\exists\ P_r \in S_u$ is scheduled at the j-th time slot of C_k $(k \neq i)$ and E_{ij} is the j-th time slot of C_i and free, we will append E_{ij} to A_i. The loop containing the predicate "If $\exists\ E_{iu} \in A_i$ such that no P_h is in S_r and satisfies $m_{xu} = P_h$ for some x" is designed to check whether any time slot of A_i can be used for scheduling P_k. We illustrate how this algorithm works by using Example 2.

$$M = \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_9 & \\ P_6 & P_{10} & P_{11} & P_{14} & P_{12} & \\ P_7 & P_8 & & P_5 & P_{13} & P_{15} \end{bmatrix}$$

Data set	S_1	S_2	S_3	S_4	S_5	S_6
Access time	4	3	4	5	5	6

$$\textit{Total access time} = 4*35+3*25+4*20+5*8+5*7+6*5=400$$

The broadcast matrix generated by using the S-NV strategy is shown as above. Similar to the SVV strategy, the data pages of S_1, S_2, and S_3 can be scheduled over C_1, C_2, and C_3, respectively. Now we discuss how the data pages of S_4 are scheduled. First, the server schedules P_9 at the fifth time slot of C_1. Then other data pages of S_4 should be broadcast on C_2. Because P_2 belongs to S_4 and has been scheduled at the second time slot of C_1, the second time slot of C_2 causes the conflicts between the data pages of S_4. So the second time slot of C_2, i.e. E_{22}, is appended to A_2. Hence, P_{11} and P_{14} are scheduled at the third and fourth time slot of C_2. Next, P_{12} of S_5 can be scheduled at the fifth time slot of C_2. Finally, the server schedules P_5, P_{13}, and P_{15} on C_3. Because the third time slot of C_3 causes that P_5 conflicts with P_{11}, the server will not schedule any data page at the third time slot of C_3. So P_5, P_{13}, and P_{15} are scheduled at the fourth, fifth, and sixth time slot of C_3. The total access time indicates that the SNV strategy is the best. In the SNV strategy, some time slots, e.g. the third time slot of C_3 and the sixth time slot of C_1, may be unused, however, these slots can be used to broadcast additional information such as indexes, updates, or invalidations; or even for extra broadcasts of extremely important pages. Moreover, the unused slots is generally only a small fraction of the total number of slots.

5 Conclusions and Future Work

In a wireless information system, each MC's desired data may access more than one data page. The affinity between data pages complicates the tasks of scheduling the broadcast data. Moreover, as a number of wireless channels are available, the design of scheduling strategies becomes more difficult. In this paper, we discuss how to schedule the broadcast data on multiple channels as each MC may access more than one data page. Five scheduling strategies were proposed to minimize the access time for the MCs. By using an example to illustrate, we see that the set-based strategies outperform than the page-based strategies. And the SNV strategy is the best because it efficiently avoids to broadcast the data pages of a data set at the same time slot.

Two extensions of this work are possible. First, in this paper we only illustrate an example to compare their efficiency of all proposed strategies. An exhaustive and accurate performance evaluation needs to be made to compare the proposed strategies. Secondly, we would like to design the scheduling strategies for a real-time environment. In a real-time wireless information system, minimizing the average access time is no longer the main performance criterion. Rather, how to guarantee that the deadlines associated with requests can be met becomes the overriding concern. Moreover, as a number of wireless channels are available, where the proper positions the data pages are in the schedule is a more complicated issue to consider.

References

[AAFZ95] Swarup Acharya, Rafael Alonso, Michael Franklin, and Stanley Zdonik, "Broadcast Disks : Data Management for Asymmetric Communication Environments", *Proceedings of the 1995 ACM SIGMOD Conference*, San Jose, California, May 1995, pp. 199-210.

[AFZ95] Swarup Acharya, Michael Franklin, and Stanley Zdonik, "Dissemination-based Data Delivery Using Broadcast Disks", *IEEE Personal Communications*, Vol. 2, No. 6, December 1995, pp. 50-60.

[CK99] Yon Dohn Chung and Myoung Ho Kim", "QEM : A Scheduling Method for Wireless Broadcast Data", *Proceedings of the Sixth International Conference on Database Systems for Advanced Applications*, HsinChu, Taiwan, April 1999, pp. 135-142.

[FZ97] Michael Franklin and Stan Zdonik, "A Framework for Scalable Dissemination-Based Systems", *International Conference Object Oriented Programming Languages Systems(OOPSLA '97)*, Atlanta, GA, October 1997.

[IV94] T. Imieliński and S. Viswanathan, "Adaptive Wireless Information Systems", *Proceedings of Special Interest Group in Database Systems(SIGDBS) Conference*, Tokyo, Japan, October 1994.

[IVB97] T. Imieliński, S. Viswanathan, and B. R. Badrinath, "Data on Air : Organization and Access", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 9, No. 3, May 1997, pp. 353-372.

[KL96] Chih-Horng Ke and Chiang Lee, "Minimizing Expecting Time in Broadcast-Based Mobile Computing Environments", *Proceedings of 1996 Workshop on Distributed System Technologies and Applications*, May 1996, Tainan, Taiwan, R.O.C. pp. 287-297.

[TO98] Kian-Lee Tan and Beng Chin Ooi, "On Selective Tuning in Unreliable Wireless Channels", *Data and Knowledge Engineering*, Vol. 28, No. 2, 1998, pp. 209-231.

[TY97] Kian-Lee Tan and Jeffrey X. Yu, "A Dynamic Scheduler for the Infinite Air-cache", *Data and Knowledge Engineering*, Vol. 24, No. 1, October 1997, pp. 97-112.

[TYE99] Kian-Lee Tan and Jeffrey X. Yu and Pin-Kwang Eng, "Supporting Range Queries in a Wireless Environment with Nonuniform Broadcast", *Data and Knowledge Engineering*, Vol. 29, No. 2, February 1999, pp. 201-221.