

December 2004

# Will It Rain Profit With Broadcast Clouds?

Aslihan Celik  
*Santa Clara University*

JoAnne Holliday  
*Santa Clara University*

Ping Ding  
*Santa Clara University*

Follow this and additional works at: <http://aisel.aisnet.org/amcis2004>

---

## Recommended Citation

Celik, Aslihan; Holliday, JoAnne; and Ding, Ping, "Will It Rain Profit With Broadcast Clouds?" (2004). *AMCIS 2004 Proceedings*. 340.  
<http://aisel.aisnet.org/amcis2004/340>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2004 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# Will It Rain Profit With Broadcast Clouds?

**Aslihan Celik**

Operations and Management Information Systems,  
Santa Clara University  
acelik@scu.edu

**JoAnne Holliday**

Computer Engineering, Santa Clara University  
jholliday@scu.edu

**Ping Ding**

Computer Engineering, Santa Clara University  
pding@scu.edu

## ABSTRACT

Wireless data providers send data such as stock prices, news headlines, traffic and weather information, and advertisements. Since wireless bandwidth is a constrained resource that determines the cost of the provider, it must be used efficiently. To conserve the bandwidth, data broadcasting techniques could be used. Existing research in data broadcasting has addressed the problems of what to broadcast and how to schedule the broadcasts. However, these methods do not focus on the bandwidth cost, rather they aim to minimize the cost of the client. In this paper, we propose the concept of “broadcast clouds” that also uses broadcasting but considers the data provider’s cost. Our approach bunches together wireless cells based on a bandwidth saving principle, and prepares a common broadcast for these. This would reduce the bandwidth cost and presumably increase the service provider’s profit. We present a heuristic to form the broadcast clouds and measure its performance via simulation. Our approach is particularly advantageous when the data items have locality and the clients are mobile and changing cells.

## Keywords

Data broadcasting, mobile computing, scheduling, wireless networks, data item locality, location-based data

## INTRODUCTION

With the proliferation of mobile computing, wireless data services have become a reality. Users of wireless devices have access to data including stock prices, news headlines, advertisements, traffic and weather information. The data are provided by either a wireless network operator (such as T-mobile or Cingular) or an independent third party that leases the wireless bandwidth from the network operator. The data service could be either bundled with the service set of the client, or purchased separately from the provider. The clients *subscribe* to this service by entering into a contract that specifies the duration of subscription and the set of data items to be delivered. However, some issues need to be resolved before these services become more prevalent. One primary concern is how to price the wireless data service such that the data provider’s revenue is higher than the costs, particularly that of the wireless bandwidth that must be leased and/or shared with other applications.

The widespread adoption of the wireless data services is hampered by the relatively limited and costly wireless bandwidth. Cellular network companies have paid huge premiums to license rather narrow frequency bands that they use to carry telephony and Internet-related data. In the case of the unlicensed Wi-Fi (i.e., IEEE 802.11) access, the *hotspots* lack the bandwidth to support several users. Therefore, a data provider is genuinely interested in reducing the bandwidth usage, and yet serve its clients.

To serve more clients with less bandwidth, data broadcasting techniques were proposed. These techniques prepare a *data broadcast* by appending individual information items together, and send it to a common channel where clients download. The clients filter the downloaded broadcast for the information that they are interested in. This solution is efficient since it eliminates the need to send the same item multiple times when each item is used by many clients.

Surprisingly, data broadcasting has not been readily adopted by wireless data providers. We have found the following shortcomings in existing broadcasting approaches that could explain this lack of interest:

1. *Existing broadcasting techniques are focused on lowering the cost of the clients, and not of the data provider.* Traditional measures of performance include Access Time (time a client has to wait until it downloads its item of interest), and Tuning

Time (time the client actually spends downloading the item – this is a measure of the energy spent). However, the cost of the provider will determine whether or not this service will be offered in the first place. The cost of the bandwidth should also be considered in designing new techniques, and included in the performance studies.

2. *Existing approaches do not distinguish between the needs of individual cells and are very unspecific about how their protocols work in a multi-cell environment.* In general, they may disseminate the same broadcast to the entire wireless network. We call this approach *one-for-all (1FA)*. Some protocols may be designed to suit the needs of a single cell, thus preparing and disseminating an individual broadcast for each cell. We call such approaches *one-for-each (1FE)*.

3. *In a one-for-all broadcast, it is highly likely that a client is interested in only a small subset of the items in the broadcast.* Most broadcast protocols show that a client is subject to a rather large broadcast with the majority of the items not of interest to the client. This effect is compounded when certain data items have “locality”. For example, the traffic on a busy intersection might be requested very frequently within 10 miles of that intersection, and is of very little interest to those travelling 100 miles from that area. Hence, with a broadcast protocol that does not consider locality of data items, many clients will have to filter through numerous data items, increasing the energy expenditure of the clients.

4. *One-for-each techniques do not fully consider the mobility needs of a client when preparing the broadcasts.* One approach frees up the broadcast space by excluding the items for the mobile hosts that have left the cell. Normally, these mobile units restart the process of requesting their items at the new cells, and have to wait until their items are included in the following broadcast intended for that cell only. A preemptive technique that would include these items in the new cell *before* the client requests them would reduce the need to resubscribe and save wireless bandwidth.

Therefore, in order for data providers to adopt data broadcasting and profit from it, it is important to develop methods that reduce the wireless bandwidth cost in the face of client mobility and data item locality. We propose the *Broadcast Clouds (BC)* approach with the goal of minimizing the bandwidth cost. *BC* is a hybrid of one-for-each and one-for-all. It groups a set of neighboring cells into clusters based on the locality of the data items, and sends the cells in a cluster the same broadcast. This way, the clients avoid resubscribing to the broadcast every time they switch cells, and the size of the broadcasts would reduce since the broadcast will be specifically prepared for that cluster. We propose a heuristic to form these clusters based on a cost saving principle. Our results indicate that the bandwidth cost under *BC* is almost always bounded by that of either *1FA* or *1FE*. Therefore, the *BC* approach saves costs particularly when the network is dynamic.

This paper is organized as follows: we first review the related literature. Next, we provide preliminaries for data broadcasting. We then describe the Broadcast Clouds approach and propose a heuristic to form the broadcast clouds. Next, we analyze the efficacy of our technique via simulations and conclude the paper.

## LITERATURE REVIEW

A central topic of this paper is broadcast scheduling. Broadcast scheduling refers to organizing a set of data items into a broadcast that is delivered to a common channel. Clients listen to that channel and download the broadcast and retrieve their items of interest. Two common goals are minimizing the *Access Time*, time until a client downloads its data item of interest, and *Tuning Time*, time a client actively listens to the broadcast channel. Research on broadcast scheduling distinguishes between two relevant classes of problems: (a) whether the scheduling is done *on-line* or *off-line*, and (b) whether the system employs *push* or *pull*. *Off-line* scheduling prepares broadcasts based on *estimates* of the user requests before the system starts. A *push* system does not rely on client requests in preparing the broadcasts (but relies on the estimates), whereas a *pull* system relies solely on the requests that the clients make. Researchers have addressed most combinations of these two classes.

Hameed and Vaidya observe that on-line broadcast scheduling is related to the problem of fair queueing. Su, Tassiulas and Tsotras derive the characteristics of an optimal on-line schedule using access probabilities. Aksoy and Franklin compare various on-line scheduling algorithms for the pull strategy. Kenyon and Schabanel show that the on-line broadcast scheduling problem to minimize average response time (access time) and the cost of broadcast is NP-hard when data items have nonuniform transmission rates in a push-based system. Guo, Das and Pinotti propose a hybrid broadcast scheduling algorithm that combines push and pull strategies. The now classic paper by Acharya et al, describes the Broadcast Disks (BD) protocol. BD orders the data items from most popular to least popular and groups the items with similar popularity together and assigns each group a “disk” which spins at a speed proportional to the popularity. The BD approach is also an on-line schedule. On the off-line broadcast scheduling side, Erlebach and Hall prove that the version of broadcast scheduling where a server can transmit one message of a given set at each time-step, answering previously made requests for that message, is NP-hard.

Our paper differs from existing literature in that we propose to schedule multiple broadcasts across the entire wireless network. We use an off-line scheduling approach combined with push-based broadcasting as well as point-to-point when it is

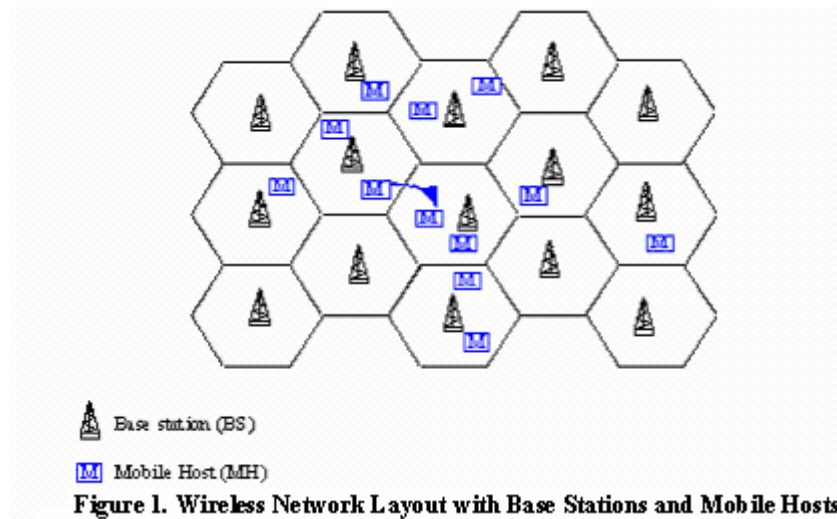
cost effective. We also use multiple channels for delivering the broadcasts. Due to mobility of the clients, our model preemptively includes some extra data items in the broadcasts in anticipation of client requests. Therefore, the information provider can only *estimate* where the requests will be generated at. To the best of our knowledge, there has not been prior research on this particular topic.

## DATA BROADCASTING IN THE WIRELESS ENVIRONMENT

In this section, we provide preliminaries for data dissemination using data broadcasting and lay out the architecture of the wireless and the supporting network environment.

### Network Architecture

We consider a basic wireless environment that is composed of mobile hosts (*MH*), base stations (*BS*), a broadcast server (*BRS*), and a subscription server (*SS*). Adhering to the standard cellular network topology, the geographical area is divided into cells. Each base station is assigned to a wireless cell and is in charge of providing the communication with the mobile hosts in that cell. Mobile hosts roam freely in between cells. We adopt the standard hexagonal representation of the wireless cells. Thus each cell has six neighbors as shown in Figure 1.



**Figure 1. Wireless Network Layout with Base Stations and Mobile Hosts**

### Data Broadcasts

Each mobile client subscribes to the broadcasting and choose its items by sending a request to the *SS*. The clients will keep downloading their items of interest from successive broadcasts.

The *BRS* is in charge of collecting client requests, and preparing the broadcasts according to the broadcasting protocol. The *BRS* encrypts each data item, and gives the decryption keys to its subscribers only. Once a broadcast is ready, copies of it are sent to the base stations which disseminate them to the clients in their cells. The server repeats this process by continuously preparing and sending broadcasts. The mobile units tune into the broadcast, and retrieve the item(s) that they are interested in. The base station (*BS*) is in charge of keeping track of the clients in its cell, and coordinating hand-offs with the neighboring cells. When a client wants to subscribe to the data service, it contacts the subscription service.

## THE BROADCAST CLOUDS TECHNIQUE

The *Broadcast Clouds (BC)* technique, seeks a balance between sending the same broadcast to the entire network (one-for-each) and sending an individual broadcast to each cell (one-for-all). In designing *BC*, our goal is to minimize the bandwidth use, yet achieve an acceptable level of quality of service. We note that many wireless cells are very small and may not differentiate much in terms of data locality. This is particularly true for adjacent cells. We propose to group these adjacent cells and prepare a common broadcast for the group. Such a group of cells is called a *cloud*. The number of cells in a cloud is a function of the probability distribution of the demand for the data items, the number of clients in each cell, and the movement pattern in and out of each cell. Note that a cloud may consist of a single cell. In the following, we define preliminaries for constructing the clouds.

### Item Locality

We represent the locality of a data item  $i$  by a probability density function  $f_i(x,y)$  in the  $x$  and  $y$ - coordinates on the plane, and the cumulative probability function:

$$F_i(x_1 < x < x_2, y_1 < y < y_2) = \int_{y_1}^{y_2} \int_{x_1}^{x_2} f_i(x,y) dx dy$$

$$0 \leq F_i(x,y) \leq 1$$

$$-\infty \leq x, y \leq \infty$$

This function represents the frequency of requests for an item  $i$  in a given region delimited by  $[(x_1, y_1)(x_2, y_2)]$

Note that this density function may also be discrete, and can be estimated from historical data.

### Client Mobility

We now define and derive terms for modeling the bandwidth costs due to client mobility. Recall that when the clients switch cells, they incur a subscription cost as part of the hand-off process.

A cloud  $I$  is said to be *adjacent* to another cloud  $J$  if a cell in  $I$  is adjacent to a cell in  $J$ . Let  $r_{I,J}$  be the rate of clients moving from cloud  $I$  into cloud  $J$ . The rate is in terms of percent of clients residing in a cell. The total number of clients in the network,  $NumClients$ , is assumed to be constant.

Let the number of clients in cloud  $I$  be  $NumClients_I$ , and let this number be constant over time. During a broadcast, the number of clients moving from cloud  $I$  into another cloud  $J$  is defined as,

$$M_{I,J}(T_B) = NumClients_I \times r_{I,J} \times T_B$$

where  $T_B$  is the duration of an individual broadcast. We assume that the system is in a steady-state with

$$\sum_{\forall J} M_{I,J}(T_B) = \sum_{\forall J} M_{J,I}(T_B), \text{ such that } J \text{ is adjacent to } I$$

### Broadcast Organization in Broadcast Clouds

In this section, we describe the organization of the broadcasts and derive the cost metrics to assess the size and content of the broadcast clouds.

The broadcast size,  $F$ , is fixed and is the same for all clouds. Therefore, only  $F$  items can fit in a broadcast. For items that do not fit in the broadcast but have subscribers, two options are available: (1) a separate, more expensive broadcast channel, and (2) unicasting (i.e. point-to-point delivery). Unicasting individually sends each item to its subscriber. The BC includes the most popular items in the regular broadcast to inconvenience fewest clients. Below, we suggest a technique for organizing the broadcasts and deciding which method(s) of delivery to use.

Let  $NumSubs_a$  denote the total number of subscribers for a data item  $a$ . Let  $NumSubs_a(I)$  denote the number of subscribers for item  $a$  in cloud  $I$ . We assume that

$$\sum_{\forall I} NumSubs_a(I) = NumSubs_a$$

where

$$NumSubs_a(I) = NumSubs_a \times P((x,y) \in \text{Cloud } I) = NumSubs_a \times \iint_{\text{Cloud } I} f_a(x,y) dx dy$$

We then sort the number of subscribers for each data item in cloud  $I$  and number the indices such that

$$NumSubs_1(I) \geq NumSubs_2(I) \geq \dots \geq NumSubs_F(I) \geq \dots \geq NumSubs_N(I)$$

where  $N$  is the number of data items. The first  $F$  data items will be included in the broadcast, and the rest will be either broadcast in the extra channel or individually sent (i.e., *unicast*) to their subscribers.

In order to decide which items to broadcast in the extra channel, let  $C_{ext}$  denote the cost of broadcasting using the extra channel, and  $C_{uni}$  denote the cost of individually sending an item to the client. Note that an item is unicast as many times as there are subscribers for it. Therefore, for an item  $a$ , the total cost of unicasting will be

$$C_{uni} \times NumSubs_a(I)$$

whereas the cost of broadcasting using the extra channel will still be  $C_{ext}$  for a single cell. Note that the total cost of broadcasting using the extra channel is

$$C_{ext} \times NumCells(I)$$

where  $NumCells(I)$  denotes the number of cells in cloud  $i$ . Therefore, there is a trade-off between unicasting and broadcasting: with a large number of cells and relatively few subscribers in a cloud, unicasting may be more cost effective than broadcasting. The reverse is also true. Thus, data items  $F + 1$  through  $F + e_I$  will be broadcast using the extra channel, and the items  $F + e_I + 1$  through  $N$  will be unicast. The cutoff point  $e_I$  is simply calculated using the fact that it is cost minimizing to broadcast using the extra channel when

$$C_{ext} \times NumCells(I) < C_{uni} \times NumSubs_{F+e_I+1}(I) \Rightarrow \frac{C_{ext}}{C_{uni}} < \frac{NumSubs_{F+e_I+1}(I)}{NumCells(I)}$$

Therefore,  $e_I$  is chosen such that,

$$NumSubs_{F+e_I}(I) > \frac{C_{ext} \times NumCells(I)}{C_{uni}} > NumSubs_{F+e_I+1}(I)$$

In order to perform these calculations, the BRS needs to know the expected number of subscribers for each data item in a cell. This information can be acquired from the network.

### Cost of Broadcast Clouds

Let  $C(I)$  denote the cost of bandwidth for operating a cloud  $I$  in the broadcast clouds model.  $C(I)$  is measured in monetary terms. Therefore,

$$C(I) = (B_I + A_I + H_I)$$

where

$B_I$ : cost of fixed broadcasts,

$A_I$ : cost of sending additional items not included in the broadcast, and

$H_I$ : cost of handling subscriptions that result from clients that change cells in the cloud  $i$ .

Cost of sending broadcasts,  $B_I$ , is straightforward: A cloud uses network bandwidth equal to the summation of broadcast sizes in the wireless cells. We assume that the broadcast size across the cloud is constant and the same. Therefore, the total cost of sending broadcasts is simply broadcast size times unit cost of wireless bandwidth times number of cells in the cloud.

The cost of handling subscriptions,  $H_i$ , is a function of the number of clients in the cloud and the frequency of moving in and out of the clouds. This includes hand-off cost for clients moving from cell to cell. This message traffic must be secure and reliable, meaning that a set of acknowledgement message must accompany each communication step.

Let the cost to carry out a subscription process be  $C(S)$ . Therefore, the cost to subscribe the incoming clients into cloud  $I$  from cloud  $J$  is

$$H_{J,I} = M_{J,I}(T_B) \times C(S)$$

Observe that,

$$H_I = \sum_{\forall K} H_{K,I}$$

assuming that the subscription cost is incurred for incoming clients only.

We now calculate  $A_I$ , the cost of using an extra and more expensive channel or using unicasting, to send items that do not fit in the regular broadcast. As derived previously,  $e_I$  items are broadcast using the additional channel at a cost  $C_{ext}$ , and the rest is unicast at a cost  $C_{uni}$ .

Let the number of cells in the cloud  $I$  be  $c_I$ . Therefore, the total cost of sending the additional items, where  $d$  is the size of an individual item, assuming all items are of the same size, is

$$A_I = \left( C_{ext} \times e_I \times c_I + C_{uni} \times \sum_{k=F+e_I+1}^N \text{NumSubs}_k(I) \right) \times d$$

Having derived the total cost of operating a broadcast cloud, we now turn our attention to actually forming the clouds in a manner that lowers this cost.

## CONSTRUCTING THE CLOUDS

Recall that a cloud is a collection of adjacent wireless cells that receive the same broadcast. We now derive the cost of implementing the Broadcast Clouds approach in terms of the wireless bandwidth, and propose a practical cost saving heuristic to form the clouds.

### Merging Rule

We introduce the operation *Merge*:

$$\widehat{M}(I \bullet J) \rightarrow \widehat{IJ}$$

that merges two individual cells or clouds  $I$  and  $J$  into a single cloud  $IJ$ . The cells in this new cloud receive the same broadcast. The following *Merging Rule* states that two clouds should be merged if it is more expensive to operate them individually than in the merged mode.

**Merging Rule:** Given two neighboring clouds,  $I$  and  $J$ , merge them if  $C(I) + C(J) > C(\widehat{IJ})$  where  $C(\widehat{IJ})$  is the cost for the merged clouds.

The cost of sending the broadcasts is the same before and after merging the clouds since two broadcasts must reach the two clouds in exactly the same way, and the size of the broadcast is fixed. The other cost components vary according to the network parameters.

$c_{\widehat{IJ}}$  is the number of cells in the new cloud.

Since movements within a cloud are cost-free, the cost components are as follows:

$$A_{\widehat{IJ}} = C_{ext} \times e_{\widehat{IJ}} \times c_{\widehat{IJ}} + C_{uni} \times \sum_{k=F+e_{\widehat{IJ}}+1}^N \text{NumSubs}_k(\widehat{IJ})$$

$$H_{\widehat{IJ}} = H_I + H_J - (H_{I,J} + H_{J,I})$$

**Lemma.** For two neighboring clouds,  $I$  and  $J$ , merge the clouds if

$$H_{I,J} + H_{J,I} > A_{\widehat{IJ}} - (A_I + A_J)$$

**Proof.** Substitute terms and perform simple arithmetic.

The lemma states that two clouds should be merged if handling hand-offs and subscriptions is more expensive than operating the additional broadcasting modes.

We can now utilize the merging rule to decide whether two clouds  $I$  and  $J$  should be merged. Next, we propose how the merging rule can be applied in a structured manner to form the clouds in the wireless network.

### A Heuristic for Constructing the Clouds

We present the heuristic MERGE-CELL to construct the clouds. The heuristic employs the merging rule discussed above in Section .

**initialize:** Form the adjacency graph of the network. Each node represents a wireless cell. There is an edge between two adjacent cells. Label the central node as 1. Label the node north to 1 as 2, and increment in clockwise fashion. When node 2 is reached, label the node north to 2 as 8 and continue. The labeling produces concentric hexagons and is illustrated in Figure 2.

**step k=1.** Start at node 1, make it a cloud  $d_1$ .

**step k=2.** For node 2, define

$$d_{12} = \widehat{M}(2 \bullet d_1)$$

and  $d_2$ , cloud with node 2 only. Compute,

$$S = C(d_1) + C(d_2) - C(d_{12})$$

If  $S < 0$ , merge 2 with  $d_1$ , else make node 2 a cloud,  $d_2$ . Increment k by 1.

**step k=i.** For node  $i$ ,  $i \leq N$ , compute

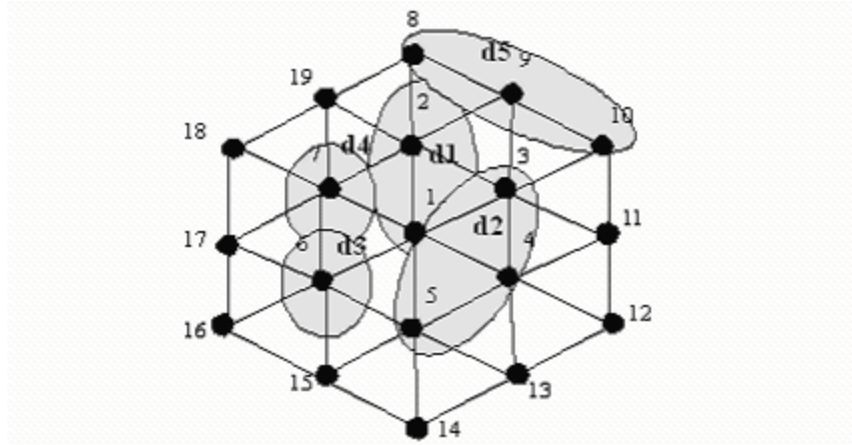
$$\min \{S_j = C(d_j) + C(d_{M+1}) - C(d_{ji})\} \forall j$$

such that  $j$  is adjacent to  $i$ , for  $j < i$ , and  $M$  is the maximum cloud index used so far, and  $d_{M+1}$  stands for the cloud formed by node  $i$  alone.

This finds the minimum of the cost of merging  $i$  with any existing adjacent cloud and the cost of making  $i$  a new cloud. Merge  $i$  with the cloud that yields the minimum cost, or make  $i$  a cloud,  $d_{M+1}$  if  $\min S_j > 0$ .

**example:** Decision for node  $i=11$  is shown in Figure 2. Shaded areas are existing clouds. Compute

$$\min \{S_2 = C(d_2) + C(d_6) - C(d_{2,11}), S_5 = C(d_5) + C(d_6) - C(d_{5,11})\}$$



**Figure 2. Example for Merge-Cell heuristic: decision point for Cell 11.**

where  $d_6$  is a cloud for node 11.

If  $s_2$  is the minimum, merge 11 with  $d_2$ ; if  $s_5$  is the minimum, merge 11 with  $d_5$ ; if both  $s_2$  and  $s_5$  are positive, then construct a new cloud  $d_6$  for cell 11.



## PERFORMANCE ANALYSIS

To analyze the performance of the Merge-Cell heuristic, we simulated the one-for-each (1FE), the one-for-all (1FA), and the Broadcast Clouds (BC) techniques. 1FE is the extreme where there is a cloud for each single node, and 1FA is the other extreme with one cloud for the entire network.

To simulate locality, the data items are randomly assigned to sets of neighboring cells. For each data item, a center cell is randomly chosen, and a given number (i.e., *NumLayers*) of layers of cells around the center cell are also modeled to have subscribers for that data item. We call this set of cells a *neighborhood*.

When clients switch cells, they incur subscription costs if the new cell is not part of the same broadcast. We do not model clients individually; we rather model their collective moves at the assumed steady-state.

The broadcasts are prepared and disseminated as described previously. The additional broadcasting channel is more expensive, and unicasting is cheaper than the broadcast channel. We assume that the cost for the regular broadcasting channel to transmit 1 bit is  $C_B$ . We define the costs of other channels in multiples of  $C_B$ .

The simulation parameters and their baseline values are shown in Table 1. Note that the number of available items is rather small, but relative to the available broadcast size, it is significant: In the baseline, 50% of all items can fit in a broadcast.

PARAMETER	DESCRIPTION	BASELINE VALUE
NumClients/cell	Number of clients per cell	10-100
NumCells	Number of cells	91
F	Fixed broadcast size, in number of items	5
NumItems	Number of available items	10
M_Rate	Mobility rate: rate of clients moving from a cell to one neighbor cell	10%/hour
C_B	Cost of broadcasting in the regular broadcast channel	$\$C_B/\text{bit}$
C_S	Cost of wireless channel used for subscription	$\$10C_B/\text{bit}$
C_ext	Cost of the extra broadcast channel	$\$2C_B/\text{bit}$
C_uni	Cost of unicasting	$\$0.5C_B/\text{bit}$
Subsize	Size of subscription packets (including encryption)	10KB
num layers	number of layers of cells in a neighborhood	2 (19 cells)
Bcast_rate	Broadcast rate	19.2Kbps
itemsizes	Size of a data item	1KB

**Table 1. Parameters and baseline values for the simulation**

### Baseline Results

The simulations are run with the same random data item distributions for all three techniques. The results of the simulation with the baseline parameters are presented in Figure 3.

All three broadcasting techniques incur  $F \times \text{numcells}$  many fixed broadcasts which cost  $\$3,727,360C_B$ . Only the *additional cost* due to extra broadcast channel, unicasting and hand-off is represented on the y- axis. The baseline results indicate that, with the given parameters, the BC approach outperforms 1FA and 1FE in terms of the total cost. The additional cost the BC approach incurs grows slower than those of the 1FA and 1FE, and it contributes to the *total* cost up to 21% only, whereas in 1FE it contributes to up to 50%.

It is not meaningful to discuss Access Time in this context since our broadcast design guarantees that all clients receive their items of interest within the duration of a broadcast. We expect that the Tuning Time would not be significantly different in the three approaches that we simulate, again due to the fact that all items are received during a broadcast.

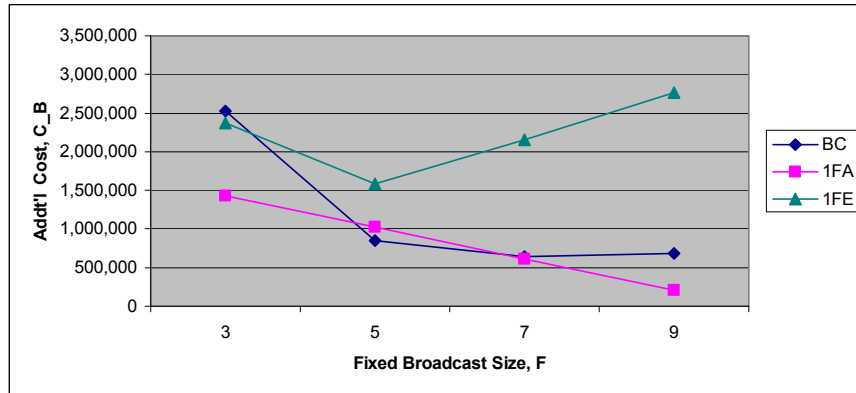


Figure 3. Baseline Simulation Results

In the following sections we vary the baseline parameters and discuss the results.

#### Sensitivity to Rate of Mobility.

We varied  $M\_Rate$  from 0 (immobile) to 20 per hour. The results, as shown in Figure 4, indicate that while the 1FA approach yields the same cost in all cases, 1FE and BC vary greatly with the mobility rate of the clients.

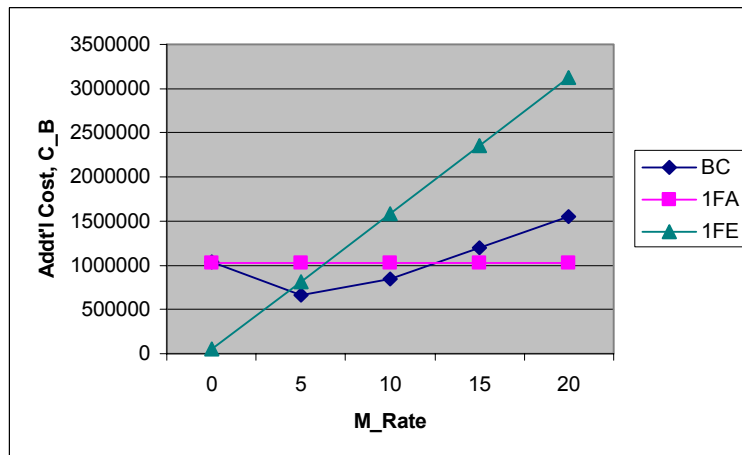


Figure 4. Sensitivity to Mobility Rate of the Clients

When  $M\_Rate=0$ , the BC constructs a single cloud with all the cells, thus overlapping with the 1FA approach. 1FE is the lowest of all since it produces very low additional cost (due to no movements between cells). As the  $M\_Rate$  is increased, 1FE moves up, while BC first moves down, then up. At  $M\_Rate=5$ , BC incurs less cost than 1FE. At  $M\_Rate=10$ , BC is still the best, however 1FE performs worse than 1FA. At  $M\_Rate=0$ , BC starts moving up. This is due to the cost of the inter-cell movements taking over. As  $M\_Rate$  is further increased, the same trend continues. Overall, 1FE has greater variation than the BC, thus it is more sensitive to the changes in the mobility rate of the clients.

#### Sensitivity to Broadcast Size

Figure 5 illustrates the additional cost of the three broadcasting approaches for various values for  $F$ , the fixed broadcast size parameter. As  $F$  is increased, 1FA decreases uniformly since almost all data items fit in the broadcast. At  $F=10$ , the additional cost that 1FA generates should be zero. 1FE and BC, however, start at much higher, get lower, then increase. The increasing trend in 1FE is due to increased subscription activity: As the broadcasts ( $F$ ) get longer, more clients switch cells during the broadcast, thus increasing cost due to subscriptions. BC is affected similarly, but the use of the clouds dampens the effect of

the subscription cost. At  $F=3$ , 1FE performs slightly better than BC, and 1FA is the best of all three. Between  $F=5$  and  $F=7$ , BC is the best performing approach. 1FA becomes the best approach for  $F>7$ .

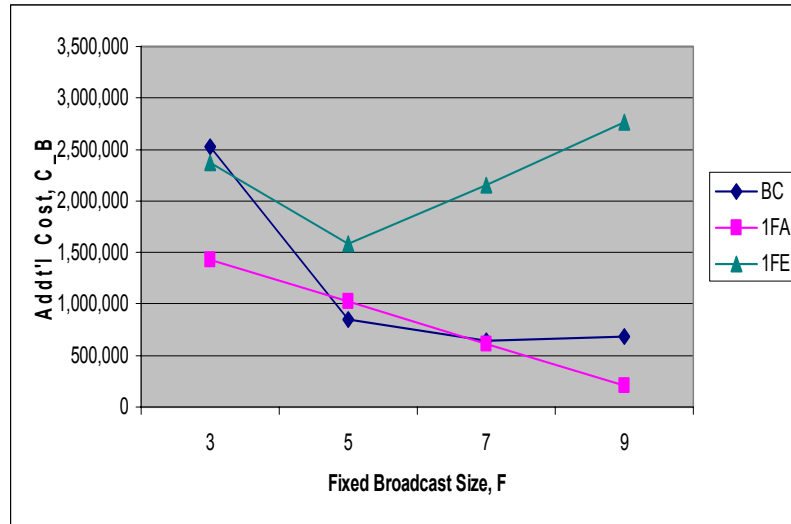


Figure 5. Sensitivity to Fixed Broadcast Size, F

#### Sensitivity to Hand-off (Subscription) Cost

In order to see the extent of its effect on the broadcasting cost, we varied the subscription cost ( $C_S$ ) parameter. Figure 6 shows that while 1FE uniformly increases with increasing  $C_S$ , 1FA remains constant, as expected. We are more interested in the effect of the  $C_S$  on the BC. At  $C_S=1$  (i.e., the subscription channel cost is the same as the broadcasting channel cost, all things being equal), 1FE performs better: it is cheap to switch cells and there is no need to consider incoming clients' potential requests in the broadcast as the BC approach does. However, as  $C_S$  is increased, its cost becomes more pronounced: both BC and 1FE costs increase. BC's cost increase is slower due to cost savings incurred by forming clouds. However, these savings are no longer enjoyed after a certain point: at  $C_S=15$ , BC's cost exceeds that of 1FA.

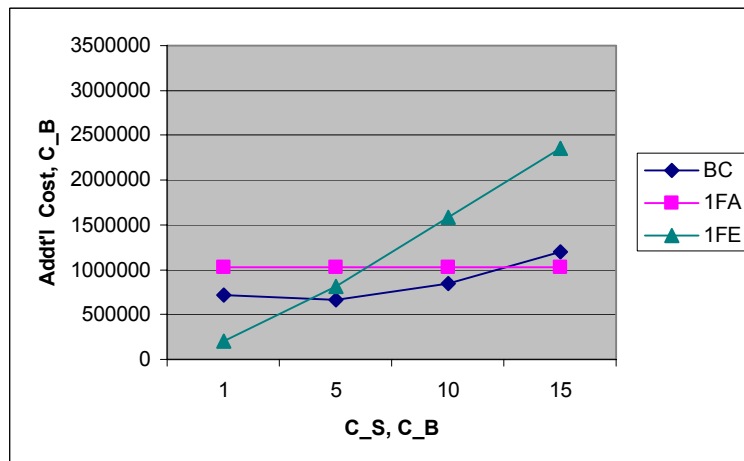


Figure 6. Sensitivity to Subscription Cost ( $C_S$ )

## CONCLUSIONS

This paper argues that a wireless data provider wants to keep the use of the wireless bandwidth at a minimum since bandwidth is a major cost component in providing the service. The bandwidth could obviously be reduced by broadcasting the data, and a number of techniques have been proposed by researchers. Although broadcasting works well, existing techniques have not concentrated on making a good enough case for data providers to adopt data broadcasting, particularly because they lack analyses that concentrate on the costs of the data provider. Rather, they have focused on the client experience.

In this paper, we addressed a long neglected problem of disseminating data broadcasts to a network of wireless cells while minimizing the bandwidth use. We first identified that broadcasts could be sent in two general formats: one for each cell (1FE) and one for the entire network (1FA). We noted that some data items could exhibit *locality*, i.e., a higher concentration of requests around certain geographic locations. We then proposed the Broadcast Clouds (BC) approach to find the optimal grouping between the extremes 1FA and 1FE, and a heuristic to form the *clouds* based on a cost-saving principle. We then compared the BC with the 1FA and the 1FE broadcasting approaches using a simple broadcast scheduling technique. Our results indicate that while each of the three approaches has strengths and weaknesses, the cost of BC is almost always bounded by that of either 1FA or 1FE. Therefore, the BC offers a nice alternative particularly when the network parameters are dynamic.

Our additional contribution is an understanding that the wireless bandwidth cost has an effect on the broadcast dissemination, which in turn affects all aspects of a data broadcasting application, including broadcast scheduling (i.e., how to organize the broadcasts). Traditional performance metrics - the access and tuning times - though worthwhile, are not sufficient for optimizing the overall network cost because they consider only the needs of the clients and not those of the data provider. Indeed, if the data provider can't make a profit, the service will not be offered.

## ACKNOWLEDGMENTS

We thank all of the anonymous reviewers and conference co-chairs for their valuable comments.

## REFERENCES

1. Acharya, S., R. Alonso, M. Franklin and S. Zdonik. "Broadcast Disks: Data management for asymmetric communication environments." *Proceedings of ACM SIGMOD, San Jose, CA, May (1995)*.
2. Aksoy, D. and M. Franklin. "RXW: A scheduling approach for large-scale on-demand data broadcast." *IEEE/ACM Transactions on Networking (1999) 7(6):846-860*.
3. Datta, A., D. VanderMeer, A. Celik and V. Kumar. "Adaptive Broadcast Protocols to Support Efficient and Energy Conserving Retrieval from Databases in Mobile Computing Environments." *ACM Transactions on Database Systems (1999) 24(1):1-79*.
4. Erlebach, T. and A. Hall. "NP-hardness of broadcast scheduling and inapproximability of single-source unsplittable min-cost flow." *Proceedings of 13th Annual ACM-SIAM Symposium on Discrete Algorithms, San Francisco, CA, (2002), pp 194-202*.
5. Guo, Y., S. K. Das and C. M. Pinotti. "A new hybrid broadcast scheduling algorithm for asymmetric communication systems: Push and Pull data based on optimal cut-off point." *Proceedings of ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM 2001), July 2001, Rome, Italy, pp 123-130*.
6. Hameed S. and N. H. Vaidya. "Efficient algorithms for scheduling data broadcast." *Wireless Networks 5 (1999) pp 183-193*.
7. Imielinski, T., S. Viswanath and B. R. Badrinath. "Data on air: Organization and access." *IEEE Transactions on Knowledge and Data Engineering, 9(3):353-372, May/June (1997)*.
8. Kenyon, C. and N. Schabanel. "The data broadcast problem with non-uniform transmission times." *Proceedings of ACM-SIAM Symposium on Discrete Algorithms, January (1999), Baltimore, MD, pp 547-556*.
9. Su C.-J., L. Tassiulas and V. J. Tsotras. "Broadcast scheduling for information distribution." *Wireless Networks 5 (1999) pp 137-147*.