Boosting the Performance of an Adaptive Push System in Environments with Locality of Demand

P. Nicopolitidis, G. I. Papadimitriou, *Senior Member, IEEE*
and A. S. Pomportsis, *Member, IEEE*

Department of Informatics
Aristotle University of Thessaloniki, Box 888
54124, Thessaloniki, Greece.
e-mail: petros@csd.auth.gr

***Abstract*— In many data broadcasting applications, clients are grouped into several groups, each one located in a different region, with the members of each group having similar demands. This paper proposes a mechanism that exploits locality of demand in order to increase the performance of wireless data dissemination systems. Specifically, it trades the received energy per bit redundancy at distances smaller than the radius of the service area for an increased bit rate for transmission of items demanded by clients at such distances. This results to an increased transmission speed for many items. The bit rate for an item transmission is dynamically determined from the distance between the server's antenna to the group of clients that demand this item via a simple feedback from the clients. Simulation results are presented that reveal significant performance improvement over fixed bit rate broadcasting in environments characterized by locality of client demands.**

Keywords: **Adaptive Data Broadcasting, Learning Automata, Locality of Demand, Variable Bit Rate.**

I. Introduction

Data broadcasting has emerged as an efficient means for the dissemination of information over asymmetric wireless networks [1]. Examples of data broadcasting applications are traffic information, weather information and news distribution systems. In such applications, client needs for data items are usually overlapping. Consequently, broadcasting stands to be an efficient solution, as the broadcast of a single information item will likely satisfy a (possibly large) number of client requests. Moreover, in many applications, such as weather information and news distribution, the locations of clients determine their demands.

Communications asymmetry is due to a number of facts, such as asymmetry in equipment (e.g. lack of client transmission capability, client power limitations), asymmetry in the network system (e.g. small uplink/downlink bandwidth ratio) and application asymmetry (e.g. traffic pattern of client-server applications).

So far, three major approaches have appeared for the server's broadcast program:

- In the pull-based approach (e.g. [2]), the server broadcasts information after explicit requests made by the mobile clients via the uplink channel. This approach is able to adapt to dynamic client demand patterns, however it is inefficient from the point of view of scalability. This is because when the client population becomes too large, the client requests will either collide with each other or saturate the server.
- In the push-based approach (e.g. [3], [4], [5]), the server is assumed to have an a-priori estimate of the demand per-information item and makes item broadcasts according to these estimates. Push systems provide high scalability and client hardware simplicity since clients do not need to include data packet transmission capability. However push systems are unable to operate efficiently in environments with dynamic demand patterns. Nevertheless, with minimal changes to client and server hardware, [6] extends the applicability of the push approach to environments characterized by a-priori unknown and dynamic client demands and presents results that reveal efficient operation in such environments.
- Hybrid approaches (e.g. [7]) try to combine the benefits of the pure-push and pure-pull approaches. However they need to carefully strike a balance between push and pull and manage a number of additional issues (determination and dynamic selection of bandwidth available for push and pull, selection of items to be pushed and those to be pulled, etc).

Information dissemination applications can be characterized by locality of client demands. A possible example of this case could be the case of a museum possessing the necessary infrastructure in order to deliver to the users information regarding the exhibits. Most museums contain several sectors with each sector containing exhibits of a different type (e.g. Egyptian, Greek, etc.). It would be desirable for visitors within a sector to be aided in their tour by receiving information regarding the contents of the sector in their native language. Supposing that the information server broadcasts such information at several languages it can be seen that locality of demand indeed exists, as groups of visitors (which are of the same nationality) tend to be at the same place and many groups are usually present inside the museum at the same time.

In a wireless data dissemination system, the transmission power of the broadcast server determines the service area. Thus, if one wants to provide data dissemination services in an area of radius R, transmission power must be set at such a level that guarantees the necessary energy per bit to noise density per Hz (E_b/N_0) ratio for clients located at the border of the service area. However, in wireless cellular environments the path loss of wireless signals is a $1/d^n$ type loss with a typical $n \geq 4$ [8]. This fact creates an increasing redundancy in the E_b/N_0 figure for clients at distances $d < R$ from the antenna.

This paper proposes a mechanism that exploits locality of demand in order to increase the performance of wireless data dissemination systems. Locality of demand means that clients are grouped into groups each one located at a different place. Additionally, members of each group have similar demands different from those of clients at other groups. The proposed approach can trade the E_b/N_0 redundancy at clients in groups at distances $d < R$ for an increased bit rate for the broadcast of the items demanded by these groups. Knowledge of client positions is conveyed to the server via a simple feedback pulse from the clients, a mechanism that was used in [6] in order to provide adaptivity to dynamic client demands. Thus, the proposed approach is presented in the context of the adaptive wireless push system of [6].

The remainder of this paper is organized as follows: After a brief introduction to Learning Automata, Section II presents the proposed variable bit rate adaptive wireless push system. Simulation results which reveal the performance superiority of the proposed approach to that of the fixed rate adaptive wireless push scheme of [6] in environments with locality of demand are presented in Section III. Finally, Section IV summarizes and concludes the paper.

II. The Variable Bit Rate Adaptive Wieless Push System

A. Learning Automata

The aim of many intelligent systems is to be able to efficiently work in environments with unknown and varying characteristics. A solution to this problem is Learning Automata [9], [10], [11], [12], which are structures that can acquire knowledge regarding the behavior of the environment in which they operate.

A Learning Automaton is an automaton that improves its performance by interacting with the random environment in which it operates. The goal of a Learning Automaton is to find among a set of actions $a_1, a_2, ..., a_M$ the optimal one, such that the average penalty received by the environment is minimized. The operation of a Learning Automaton constitutes a sequence of repetitive cycles which eventually lead to the target of average penalty minimization. The automaton maintains $p_1(n), p_2(n), ..., p_M(n)$, which is a vector representing the probability of selecting action i at cycle n. Obviously, $\sum_{i=1}^{M} p_i(n) = 1$. For each cycle, the automaton chooses an action and receives the environmental response triggered by the selected action. Based on this response the automaton updates the probability vector $p(n)$ to $p(n+1)$and uses it determine the selection of the next action.

There exist different automata types according the nature of the environmental response. If this takes only the values 0 and 1, indicating only reward or penalty respectively, the automaton is known as a P-model one. However, due to the fact that in many cases a P-model gives only a gross estimation of the environment, schemes where environmental response can be neither completely rewarding or penalizing have been devised. These kind of Learning Automata work with environmental responses which, after normalization, lie in the interval [0..1]. In a Q-model, the environmental response can have more than two, still finite however, possible values in the interval [0..1]. In an S-model environment, the environmental response can take continuous values in [0..1].

Learning Automata have been found to be useful in systems where incomplete knowledge regarding the environment in which those systems operate exists. In the area of data networking Learning Automata have been applied to several problems, including the design of self-adaptive MAC protocols, both for wired and wireless platforms, which efficiently operate in networks with dynamic workloads [13], [14], [15], [16]. Other applications of Learning Automata include queueing systems, task scheduling, image compression, pattern recognition and telephone-traffic routing.

B. The Learning Automaton-based Broadcast Server

In the adaptive wireless push system of [6] the server is equipped with an S-model Learning Automaton [9], [10], which contains the server's estimate p_i of the demand probability d_i for each data item i among the set of the items the server broadcasts. Clearly, $\sum_{i=1}^{M} d_i = 1$, where M is the number of items in the server's database.

According to [4], at each cycle, the server selects to transmit the item i ithat maximizes the cost function $G(i) = (T - R(i))^2 \frac{d_i}{l_i}, \quad 1 \leq i \leq M$, where T is the current time, $R(i)$ the time when item i was last broadcast and l_i is the length of item i. For items that haven't been previously broadcast, $R(i)$ is initialized to -1. If the maximum value of $G(i)$ is given by more than one item, the algorithm selects one of them arbitrarily. Upon the broadcast of item i at time T, $R(i)$ is changed so that $R(i) = T$. After broadcasting item i, the algorithm proceeds to select the next item to broadcast. At the adaptive system ([6]), after the transmission of item i, the broadcast server awaits for acknowledgment from every client that was waiting item i. Such clients acknowledge reception and each one transmits a short pulse. The amplitudes of the acknowledging nodes' pulses are added at the server, which uses the aggregate received pulse strength to update the automaton. The probability distribution vector p maintained by the automaton estimates the demand probability d_i (and thus the popularity) of each information item $i, 1 \leq i \leq M$. For the next broadcast, the server chooses which item to transmit by using the updated vector p.

When the transmission of an item i does not satisfy any waiting client, the probabilities of the items do not change. However, following a transmission that satisfies clients, the probability of item i is increased. A Liner Reward-Inaction (L_{R-I}) probability updating scheme [10] is employed after the transmission of item i (assuming it is the server's k^{th} transmission):

$$p_j(k+1) = p_j(k) - L(1-b(k))(p_j(k) - a),\ \forall j \neq i \tag{1}$$

$$p_i(k+1) = p_i(k) + L(1-b(k)) \sum_{i \neq j} (p_j(k) - a)$$

In the above scheme $p_i(k) \in (a, 1)\ \forall k$ and $L, a \in (0, 1)$. If the probability p_i of an item i becomes zero, then $G(i)$ would be very close to zero as well. However, item i, even if unpopular, still needs to be seldomly transmitted since some clients may request it. Additionally, the dynamic nature of client demands might make this item popular in the future. Parameter a prevents the probabilities of non-popular items from taking values in the neighborhood of zero and thus increases the adaptivity of the automaton. $b(k)$ is the environmental response and is represented by the sum of the received feedback pulses after the server's k^{th} transmission. Upon reception, this sum is normalized in the interval [0,1]. A value of $b(k)$ that equals 1 represents the case where no client acknowledgment is received. Consequently, the lower the value of $b(k)$, the more clients were satisfied by the server's k^{th} transmission.

The above protocol needs a mechanism that will enable the server to possess an estimate of the number of clients under its coverage so as to perform the normalization procedure of the sum of feedback pulses. This can be achieved by the broadcasting of a control packet that notifies all clients to respond with a pulse. The server will use this aggregate received pulse strength to estimate how many clients are within its coverage area and use this value to perform the normalization.

However, the signal strength of each client's pulse at the server depends on its distance from the server's antenna. Moreover, it is of dynamic nature due to the mobility of the clients. Since the path loss is a $1/d^n$ type loss with a typical $n = 4$, the feedback pulse of clients located close to the broadcast server will be orders of magnitude stronger than the pulses of clients further away. In order to prevent those clients that are closer to the server's antenna from dominating the voting, we use a power control mechanism on the returning pulses. Thus, every information item will be broadcast including information regarding the signal strength used for its transmission. Due to path loss, clients far from the base station will receive the item and measure a low signal strength. Based on the received signal strength of the item and the piggybacked information regarding the signal strength at which the item was originally transmitted, the client will accordingly set the amplitude of its feedback pulse. Thus, clients acknowledging the receipt of an item will measure the item's signal strength and set the amplitude of their feedback pulse to be the inverse of the ratio (signal strength of the received item) / (signal strength of the item transmission). For example, assume that the server broadcasts all items with signal strength 1. Upon reception by a client of a item with signal strength k ($k \leq 1$), the client will set the amplitude of its feedback pulse at $1/k$. Using this form of power control, the contribution of each client's feedback pulse at the server will be the same order of magnitude and will not depend on client-server distance.

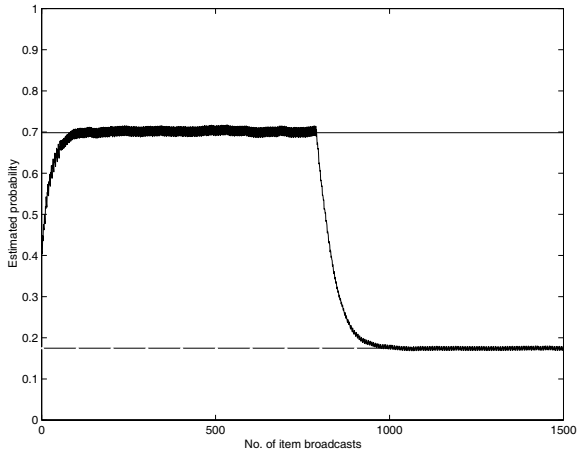

Fig. 1. Convergence of automaton estimation of the demand of an item.

Using the re-enforcement scheme of Equation (1), the item probabilities estimated by the automaton converge to the actual demand probabilities for each information item. Via simulation, this convergence is shown in Figure 1 for a randomly selected information item. Overall client demand for the item is initially unknown to the server. It can also be seen that they are of a dynamic nature as well: At some time instant, the initial overall demand probability for the selected item (solid line) changes to a new one (dashed line). It is clearly seen, that convergence of the item probability estimated by the automaton to the overall client demand for this item is achieved. Moreover, simulation results in [6] and [17] have demonstrated efficient operation in environments characterized by dynamic and a-priori unknown to the server, client demands.

C. The Bit Rate Variation Mechanism

To the authors' knowledge, locality of demand has not been taken into account in related research so far; on the contrary, clients are assumed to be uniformly distributed inside the service area and generally make item requests using the same or similar patterns (e.g. [4], [6]). In many cases however, clients are grouped into several groups located at different places with the clients of each group having similar demands, different from those of clients at other groups. In the discussion following all clients are assumed to be positioned outside the near field of the sever's antenna.

In a typical data broadcasting application (and generally in wireless cellular systems), service area is an area of

certain radius R inside which mobile clients are able to receive information items while experiencing a Bit Error Rate (BER) below or equal to a certain requirement value. The size of the service area depends on a number of parameters, such as the type of modulation that is used, the bit rate, the server's transmission power and noise density per Hz and is determined by a simple rule stating that its border is where the received energy per bit E_b divided by the noise density per Hz N_0 equals a certain constant A. The value of A is determined so that the E_b/N_0 ratio results to a BER below or equal to a set requirement. Thus at the border of the service area it stands that:

$$\frac{E_b}{N_0} = A \Rightarrow E_b = A' \tag{2}$$

where $A' = AN_0$.

Since $E_b = T_b S_R$, where S_R is the received power at distance R from the antenna and T_b is the bit duration, we can rewrite the above relation as:

$$T_b S_R = A' \tag{3}$$

Finally, since in wireless cellular environments the path loss of wireless signals at distance d is a $1/d^n$ type loss (with a typical value of $n \geq 4$), (3) can be expressed as:

$$R^{-n} T_b = A' \tag{4}$$

In fixed bit rate systems, clients inside the service area (at distance $d < R$) experience even lower BERs than those required due to smaller distance from the antenna. Thus, for such clients it holds that $E_b > A'$ and therefore $d^{-n} T_b > A'$. Assume that there exists locality of demand, as defined earlier. Then we can exploit the above mentioned redundancy in the received BER by dynamically reducing the T_b parameter for each information item i so that it always holds that $d^{-n} T_b(d) = A'$, where d is the distance of the group of clients that access item i.

Based on the above reasoning, the adaptive system of [6] is enhanced as follows: Each information item comprises a header that contains information that uniquely identifies the item. All item headers are always broadcast with the default T_b value, while the T_b value for the main item payload can be altered by the server. After the transmission of item i, the server waits for acknowledgment pulses from all mobile clients that were satisfied by this transmission. Since we consider groups of clients having the same interests, acknowledgment pulses for a certain item will be from a group of collocated clients and therefore arrive together at the server. The server monitors the time elapsed from the broadcast of item i until the aggregate pulse is received and uses this information to calculate the distance d of the group of clients from the antenna. When it broadcasts the next instance of this item the bit duration that will be used, $T_b(d)$, will be such that satisfies the requirement that $d^{-n} T_b(d) = A'$. Change of the bit duration is not a problem for the mobile client, as it can be informed of this via piggybacking of the new bit duration in the item header, which is always broadcast with the default T_b value.

As far as acknowledgment pulses are concerned, a client responds to the server via such a pulse if it demands item i and successfully receives i's header. We explain that this provides support for clients that may have broken away from the main group and are located further away from the antenna than the main group. Assume that such a client C, at a distance $d1$ receives only the header of i due to the fact that the main item payload has been transmitted with a bit rate determined by the location of the main group, which is closer to the antenna. In that case the server will receive more than one feedback pulses, one corresponding to the main group and one from C. In order to prevent C from starvation, the server will schedule the broadcast of the next instance of item i according to the feedback pulse of C (thus the client further away). This enables the client further away from the group to successfully receive item i when it is next broadcast. At the next broadcast of item i, C will successfully receive the item. However, this time C will not transmit a feedback pulse so as not to acknowledge twice reception of one instance of item i, a fact that would provide inaccurate information regarding demand for item i to the probability updating scheme.

In order to better understand the behavior of the system we present the following example. We assume a server with a database of 2 items of unit lengths, two groups of clients, A and B, with the first group always accessing the first item while the second group always accesses the second item. The radius of the service area is R. As far as group distances from the antenna are concerned, $d_A = R/2$, $d_B = R/3$. Two members of group A are away from the main group at distances $R/4$ and $3R/4$. Furthermore, the client at distance $3R/4$ (client C) makes new item requests with non unit probability. Finally, $T_b(R) = 1$, the path loss exponent n is 4 and initially all item payloads will be transmitted via $T_b = 1$. We illustrate the following five example steps of the algorithm:

Step 1: We assume that according to the selection procedure, the server decides to transmit item 2.

- Clients in group B receive item 2 and transmit their feedback pulses.
- From the time elapsed between the broadcast of item 1 and the reception of the aggregate feedback pulse of clients from group B, the server calculates the distance of group B from the antenna and schedules the next instance of item 2 to be broadcast via $T_b(R/3)$ which equals $\frac{T_b(R)}{3^4} = 1/81$.

Step 2: Next, the server broadcasts item 1.

- All clients of group A, except client C, demand and receive item 1 and transmit their feedback pulses.
- The server receives two groups of pulses and will schedule the next broadcast of item 1 to take place according to the pulse corresponding to the location further away from the antenna, thus via $T_b(R/2)$ which equals $1/16$.

Step 3: Next, we assume that the server broadcasts again item 1 via $T_b(R/2)$.

- All clients of group A, including client C, demand this item. Obviously due to the increased bit rate the item is received by all members of A except client C who only receives the header of item 1. All members of group A (including client C) acknowledge item 1.
- The server receives three groups of pulses and will schedule the next broadcast of item 1 to take place according to the location of the acknowledging client further away (client C), thus via $T_b(3R/4)$ which equals 0.31.

Step 4: Next, the server decides to transmit item 2 via $T_b(R/3)$ as calculated in step 1. For this broadcast, everything else will be the same with step 1.

Step 5: Next, we assume that the server broadcasts again item 1, this time via $T_b(3R/4)$ as stated in Step 3.

- All members of group A, including client C, demand and receive item 1.
- All members of group A, except for client C, transmit their feedback pulses. C does not transmit a feedback pulse due to the reason explained earlier.

III. Performance Evaluation

In order to assess the performance increase of the proposed variable rate system, we used simulation to compare it to the fixed bit rate system of [6]. The comparison is made in an environment characterized by client demands that are a-priori unknown to the server and location dependent.

A. Server model

We consider a broadcast server having a database of equally-sized Dbs data items. The server is initially unaware of the demand for each item, so initially every item has a probability estimate p_i of $1/Dbs$. In the fixed bit rate system, the server broadcasts all items with the same bit rate. In the variable rate system however, the server determines the bit rate to use for each item according to the proposed scheme.

B. Client model

We consider a client population of $ClNum$ clients that have no cache memory, an assumption also made in other similar research (e.g. [4] and [6]). Clients are grouped into G groups each one of which is located at a different distance from the antenna and outside the antenna's near field. Any client belonging to group $g, 1 \leq g \leq G$ is interested in the same subset Sec_g of the server's database. All items outside this subset have a zero demand probability at the client. Finally, $Sec_i \neq Sec_j, \forall i,j \in [1..G], i \neq j$, which means that there do not exist common demands between any two clients belonging to different groups.

Assume that such a subset comprises Num pages. The demand probability d_i for each item in place i in that subset, is computed according to the Zipf distribution, which is used in other papers dealing with data broadcasting as well ([3], [4], [5], [6]):

$$d_i = c\left(\frac{1}{i}\right)^{\theta}, \tag{5}$$

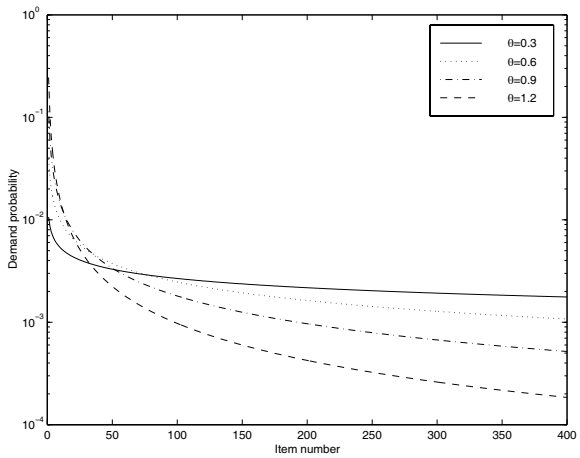

Fig. 2. Demand probability produced by the Zipf distribution for different values of θ.

$$where \quad c = 1/\sum_k \left(\frac{1}{k}\right)^{\theta},\ k \in [1..Num],$$

where θ is a parameter named access skew coefficient. For $\theta = 0$ the Zipf distribution reduces to a uniform distribution of demand for the items in the range $[k_1..k_2]$. For large values of θ, the Zipf distribution produces increasingly skewed demand patterns. The Zipf distribution can thus efficiently model applications that are characterized by a certain amount of commonality in client demands. Figure 2 shows item demands calculated via the Zipf distribution for a database subset comprising 400 items.

Client placement takes place among LP different distance points outside the antenna's near field, with the maximum distance corresponding to the coverage radius of the system. Members of a group g are initially located at a distance Loc_g . To simulate some "noise" in client locations, we introduce the parameter Dev, which determines the percentage of clients that deviate from initial client placement. For every client, a coin toss, weighted by Dev, is made. If the outcome of the toss states that the client is to deviate from the initial client placement then its position is changed to a new one selected in a uniform manner from the interval $[1..LP]$.

C. The Simulation Environment

We performed our experiments with an event-driven simulator coded in C. The simulator models the $ClNum$ mobile clients, the broadcast server and the server-client links as separate entities. We assume that the broadcast server's antenna is at the center of the circular cell and a path loss model of $1/d^n$. In order to model different group sizes, we calculated the size of each group g via the above mentioned Zipf distribution. Thus group g has a size of: $c\left(\frac{1}{g}\right)^{\theta 1}$, $where\ c = 1/\sum_k \left(\frac{1}{k}\right)^{\theta 1}$, $k \in [1..G]$.

Upon completion of a item's broadcast, the following events take place:

1) Clients that demanded the item and received its header correctly respond with a power-controlled feedback pulse. Those that demanded and received

the entire item correctly also proceed to calculate the next item to access.

2) The sum of the acknowledging feedback pulses is used by the automaton at the server to update its estimation of the item probabilities.
3) According to the received feedback, the variable rate system calculates the bit rate for the next broadcast of this item.

Assume that any client located at distance d receives items with $E_b = Th$. In the fixed bit rate system every item being broadcast is assumed to be correctly decoded at the mobile clients. As was mentioned earlier, item headers are broadcast with the default bit rate and are thus always correctly decoded by clients in the variable rate system as well. Item payloads however are correctly decoded by clients in the variable rate system if and only if they arrive at the demanding clients with an E_b figure being at least equal to Th.

The simulation is carried out until at least N requests are satisfied at each client, meaning that overall, at least $N * ClNum$ requests have been served. Finally, the overhead due to the duration of the feedback pulse and the signal propagation delay is considered to be very small compared to the item transmission time (parameter Ovh), as would happen in low-speed broadcasting applications spanning an area of several kilometers.

D. Simulation results

The simulation results presented in this section were obtained with the following parameters values: $n = 4$, Dbs=400, $ClNum$=10000, $G = 5$, $Sec_1 = [0..119]$, $Sec_2 = [120..239]$, $Sec_3 = [240..319]$, $Sec_4 = [320..359]$, $Sec_5 = [360..399]$, $LP = 100$, N=10000, Ovh=10^{-3}, L=0.15, a=10^{-4}. Figures 3-5 contain results that compare the performance of the fixed bit rate system to that of the adaptive one for different values of Dev in three different environments, Networks N_1, N_2, N_3. The parameters of the three simulation environments are:

1) N_1: $Loc_1 = 5, Loc_2 = 25, Loc_3 = 60, Loc_4 = 80, Loc_5 = 95$, $\theta 1 = 1.0$
2) N_2: $Loc_1 = 5, Loc_2 = 25, Loc_3 = 60, Loc_4 = 80, Loc_5 = 95$, $\theta 1 = 0.0$
3) N_3: $Loc_1 = 95, Loc_2 = 80, Loc_3 = 60, Loc_4 = 25, Loc_5 = 5$, $\theta 1 = 1.0$

The main conclusions that can be drawn from the Figures are:

- The performance of all schemes improves for increasing values of the data skew parameter θ. This is expected behavior [6], [17], as the Learning-Automaton adaptation mechanism manages to learn the actual demand probabilities of the various information items and use these values on the selection of the item to broadcast.
- The performance of the adaptive bit rate system is superior to that of the fixed bit rate one in all cases. This is due to the fact that in the adaptive system bit rate is not fixed but dynamically determined by client distance from the antenna; thus many items are transmitted much faster than in the fixed bit rate system resulting to the overall performance increase.
- For a non-zero value of Dev the performance of the adaptive system declines, remaining however significantly superior to that of the fixed bit rate system. This is due to the fact that as $Dev \neq 0$, not all members of a group are located at the same distance from the antenna; thus in some cases information item broadcasts for a certain client group are also acknowledged by clients further away than the location of the main group. Thus in many cases, it is the feedback pulse of the client that is furthest away that determines the bit rate to be used for the next broadcast of the same information item.
- The performance of the adaptive bit rate system is sometimes better when the sizes of groups that are located closer to the antenna are larger (e.g. performances in some cases of the adaptive system in Figure 3, when compared to with the corresponding ones in Figures 4 and 5). However even in the case

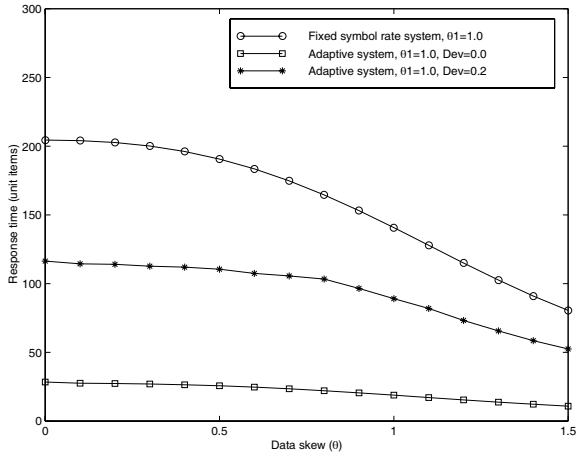

Fig. 3. Overall Mean Access Time in items versus access skew coefficient θ. Client Groups are in positions 5, 25, 60, 80, 95. $\theta 1 = 1$.

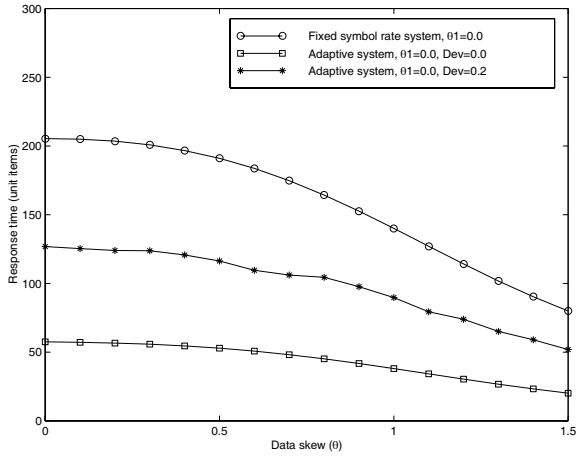

Fig. 4. Overall Mean Access Time in items versus access skew coefficient θ. Client Groups are in positions 5, 25, 60, 80, 95. $\theta 1 = 0$.

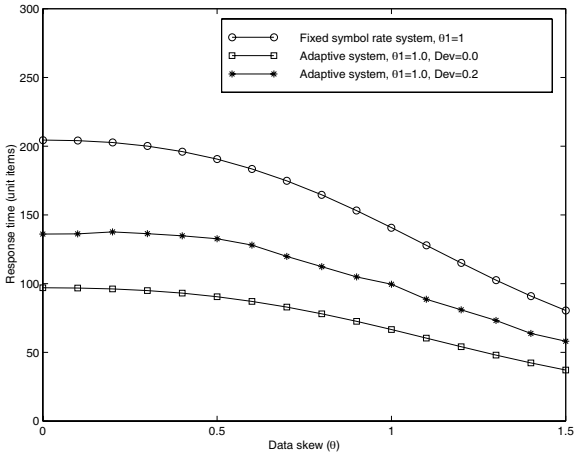

Fig. 5. Overall Mean Access Time in items versus access skew coefficient θ. Client Groups are in positions 95, 80, 60, 25, 5. $\theta 1 = 1$.

where the largest group is the one that is furthest away of the antenna (e.g Figure 5), the performance of the adaptive bit rate system is significantly better than that of the fixed bit rate one.

IV. Conclusion

With the increasing popularity of wireless networks and mobile computing, data broadcasting has emerged as an efficient way of delivering data to mobile clients having a high degree of commonality in their demand patterns. In many cases clients are grouped into several groups, each one in a different location, with the members of each group having similar demands. This paper proposed a mechanism that exploits locality of demand in order to increase the performance of wireless data dissemination systems. Specifically, it trades the E_b redundancy at a distance smaller than the coverage radius, for an increased bit rate for transmission of items demanded by client groups at this distance. Knowledge of client positions is conveyed to the server via a simple feedback from the clients. Simulation results have been presented that reveal significant performance improvement over fixed bit rate systems in environments characterized by locality of client demands.

References

[1] P. Nicopolitidis, M. S. Obaidat, G. I. Papadimitriou and A. S. Pomportsis, "Wireless Networks", John Wiley and Sons Ltd., 2003.

[2] D.Aksou, M.Franlkin, "Scheduling for Large-Scale On-Demand Data Broadcasting", In Proceedings of the IEEE Infocom, 1998, pp.651-659.

[3] S.Acharya, M.Franklin and S.Zdonik, "Dissemination-based Data Delivery Using Broadcast Disks", IEEE Personal Communications. vol.2, no.6, pp 50-60, December 1995.

[4] N.H.Vaidya, S.Hameed, "Scheduling Data Broadcast In Asymmetric Communication Environments", ACM/Baltzer Wireless Networks, vol.5, no.3, pp.171-182.

[5] C.J.Su, L.Tassiulas, "Broadcast Scheduling for Information Distribution.", In Proceedings of the IEEE Infocom May, 1997, pp.109-117.

[6] P.Nicopolitidis, G.I.Papadimitriou and A.S.Pomportsis, "Using Learning Automata for Adaptive Push-Based Data Broadcasting in Asymmetric Wireless Environments", IEEE Transactions on Vehicular Technology, vol.51, no.6, pp. 1652-1660, November 2002.

[7] K.Stathatos, N.Roussopulos, J.S.Baras, "Adaptive Broadcast in Hybrid Networks", In Proceedings of VLDB 1997, pp.326-335.

[8] J.B.Andersen, T.S.Rappaport and S.Yoshida, "Propagation Measurements and Models for Wireless Communication Channels", IEEE Communications Magazine, vol.33, no.1, pp.42-49, January 1995.

[9] K.Najim; A.S.Poznyak, "Learning Automata: Theory and Applications", Pergamon, New York, 1994.

[10] K.S.Narendra; M.A.L.Thathachar, "Learning Automata: An Introduction", Prentice Hall, 1989.

[11] G.I.Papadimitriou, "A New Approach to the Design of Reinforcement Schemes for Learning Automata: Stochastic Estimator Learning Algorithms", IEEE Transactions on Knowledge and Data Engineering, vo.6, no.4, pp.649-654, August 1994.

[12] G.I.Papadimitriou, "Hierarchical Discretized Pursuit Nonlinear Learning Automata with Rapid Convergence and High Accuracy", IEEE Transactions on Knowledge and Data Engineering, vol.6, no.4, pp.654-659, August 1994.

[13] G.I.Papadimitriou and A.S Pomportsis, "Learning Automata-Based TDMA protocols for Broadcast Communication Systems with Bursty Traffic", IEEE Communication Letters, vol.4, no.3, pp.107-109, March 2000.

[14] G.I.Papadimitriou and A.S.Pomportsis, "Self-Adaptive TDMA Protocols for WDM Star Networks: A Learning-Automata-Based Approach", IEEE Photonics Technology Letters, vol.11, no.10,pp.1322-1324, October 1999.

[15] G.I.Papadimitriou and D.G.Maritsas, "Learning Automata-Based Receiver Conflict Avoidance Algorithms for WDM Broadcast-and-Select Star Networks", IEEE/ACM Transactions on Networking, vol.4, no.3, pp.407-412, June 1996.

[16] P.Nicopolitidis, G.I.Papadimitriou and A.S.Pomportsis, "Learning-Automata-Based Polling Protocols for Wireless LANs", IEEE Transactions on Communications, vol.51, no.3, pp.453-463, March 2003.

[17] P.Nicopolitidis, G.I.Papadimitriou and A.S.Pomportsis, "On the Implementation of a Learning Automaton-based Adaptive Wireless Push System", in Proceedings of Symposium on Performance Evaluation of Computer and Telecommunication Systems 2001, pp.484-491.