AN OPTIMAL ADAPTIVE SCHEME FOR MULTIPLE ACCESS BROADCAST COMMUNICATION.

L. KLEINROCK
Y. YEMINI

Computer Science Department
University of California
Los Angeles.

0. SUMMARY.

The problem of multiple access to a broadcast channel is considered as a problem of real time distributed decision making. We assume that the decision makers are only aware of the total load on the channel when deciding access rights. This information is similar to that required by optimally controlled Slotted *ALOHA*. A new optimal access scheme under this assumption, the ***urn scheme***, is described. This scheme is shown to perform better than optimally controlled Slotted *ALOHA* and Time Division Multiple Access (*TDMA*) for all ranges of traffic load. It smoothly adapts to the load on the channel, varying from Slotted *ALOHA* at light load, ranging through an asymmetric scheduling scheme, finally eliminating collisions as it converges to *TDMA* in the heavy traffic case; thus there is no limitation on the useful capacity. The scheme is robust w.r.t. errors in decisions or information used for decisions. The control overhead is negligible and does not depend upon the system size. A variety of possible implementations is presented together with the results of analysis and simulation.

1. THE ACCESS SCHEME PROBLEM.

In what follows we consider the problem of sharing a time slotted broadcast channel by a population of bursty Packet Radio Units (PRUs) [1]. Packets are generated at each PRU according to some random laws, stored in a buffer, transmitted (using some channel access scheme) over a broadcast channel during a time slot, and finally discarded upon acknowledgment (over an assumed free acknowledgment channel) of successful reception.

A simple model of the communication channel is assumed. Problems of modulation, synchronization, coding and the like are assumed to have been solved. It is also assumed that all PRUs are synchronized to slot boundaries. Therefore the channel is a mere succession of rectangular time slots. A slot may be empty, occupied by a single packet (a success), or contain more than one packet, i.e., a "collision". A collision of two or more packets in the same slot results in a total loss. A packet which does not collide is always successful. Our main concern is the mechanism responsible for sharing the channel resource.

The problem of channel sharing is essentially a problem of decision; i.e., which PRUs should have an access right at a given slot. An ***access scheme*** is defined to be an algorithm (possibly distributed and/or non-deterministic) to determine access rights. A PRU which has both an access right and a packet ready for transmission, will transmit it.

The objective of an access scheme is to maximize the rate of successful packets (throughput). Once we specify the set of available strategies, the access scheme problem becomes that of controlling a queueing system through the service mechanism. Alas, the decision problem is non-classical, for both the observation of state and the decisions are distributed among the PRUs.

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract MDA 903-77-C-0272.

At a given time slot t, a PRU may choose between having an access right or not. This binary choice is the set of his pure strategies. It is also possible to randomize the decision between the two pure strategies, i.e., by tossing a biased coin to create a mixed strategy. Thus a busy PRU may choose, at any slot, a probability of transmission $0 \le P \le 1$, reflecting his wish to achieve an harmonious set of decisions with other members of the PRU community.

The problem of distributed decision is distinguished from classical decision theories in two respects. First, each decision maker possesses only partial state information (usually distinct from his fellows) insufficient to estimate the state. Second, the different decision makers need to coordinate their policies. In the case of an access scheme, the optimal control rule a' la classical control theory is obvious; i.e., give access rights to any SINGLE busy PRU (a PRU is said to be ***busy*** if it has a packet ready for transmission). Our problem begins where control theory ends. That is, how would the different PRUs acquire information about the identity of the busy PRUs? How would they agree upon the identity of the lucky PRU to be selected for transmission?

One possible approach to the problem of access schemes is to consider the information used for decision as a point of departure. That is, for a given amount of information, what is the best decision rule to follow?

If no information is used, the PRUs cannot adapt to the instantaneous state of the communication demands. Therefore the allocation of transmission rights must be predetermined. In an environment where the demand for the communication channel is bursty, a scheme which is not adaptive results in a severe waste of the channel. Thus adaptivity is a very desirable property to incorporate in an access scheme.

An adaptive distributed decision mechanism requires that PRUs acquire information about the state of the community and coordinate their choice of strategy. However, the processes of information acquisition and coordination may prove to be expensive in terms of both the channel overhead and the delayed decision. Moreover, an algorithm which depends on detailed state information and coordination may be highly unreliable. Thus a real time decision process must use minimal information for adaptive and reliable decisions.

Let us consider the case where the information which is used to decide access rights consists simply of the knowledge of the total number of busy PRUs at the beginning of each given slot. (We use N to denote the total number of PRUs and n to denote the number of busy PRUs). This is a case of ***homogeneous*** and ***symmetric*** information. In other words, all PRUs are supposed to use the same information; moreover, the information used for adaptivity does not distinguish between the needs of different PRUs. As far as the decision making is concerned all PRUs look the same.

Symmetric homogeneous information is assumed by some schemes to control Slotted *ALOHA* [2,3,4,5,6]. We shall later describe an information acquisition mechanism to estimate the number of busy PRUs with a negligible error and at the expense of a minimal overhead. Information about the number of busy PRUs is less (much less in terms of overhead) than perfect information (which would give the identity of each of these busy PRUs); it is more (much more in terms of the ability to adapt) than no information. What is the best decision strategy to be employed given this information?

To develop insight into the solution let us consider the case of two PRUs (N=2). If both know that the number of busy PRUs is n=1, then the obvious solution is to give both of them full access rights; only one of them will actually transmit and succeed. In this case the optimal decision (i.e., transmit with probability 1) is symmetric; both PRUs are given the same pure access right.

Now consider the case N=2, n=2; i.e., both PRUs are busy and aware of it. What is the best selection of transmission probabilities? The expected throughput S describes a surface above the square of all possible transmission probability pairs. Figure (1) depicts the throughput surface as a function of the transmission probabilities. If we restrict ourselves to symmetric policies, i.e., the main diagonal, the optimal policy is (0.5,0.5). That is, each PRU should transmit with probability 1/2. This is precisely an "optimally controlled *ALOHA* policy". We shall use this term to denote a scheme where each PRU knows n and has an access right with probability 1/n (i.e., a mixed strategy).

However, the throughput obtained by the policy (0.5,0.5) is only 0.5. This is a saddle point of the throughput surface. A better solution would have been to choose either of the extreme policies (1,0) or (0,1). That is select an arbitrary PRU and give him full transmission rights; the other should be quiet. How can this be implemented? Here we use the homogeneity. If both PRUs use the same information for decision, then they can reach a coordinated decision through a preprogrammed set of rules. In this case it is only required to have some predetermined priority mechanism to select any one of them.

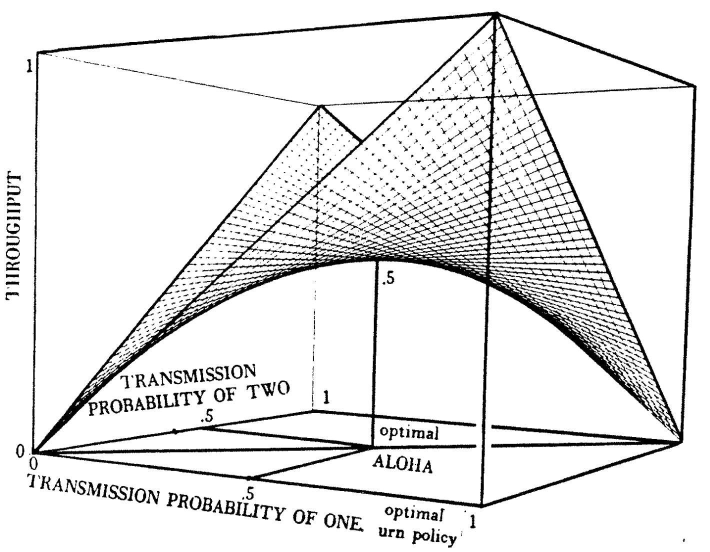


FIGURE (1): THROUGHPUT SURFACE

The optimal strategy is, therefore, ASYMMETRIC. It is possible to prove[7] that optimal strategies are always pure (extreme) strategies and therefore asymmetric. In other words some PRUs should get full transmission rights while others get none (i.e., the transmission probabilities P are 0 or 1). We shall not dwell upon the details of the proof. Rather, we describe an urn model of the problem which renders the solution intuitively clear.

Consider each PRU as a colored ball in an urn; black for busy, white for idle. The access scheme is essentially a rule to sample balls from the urn (those given full access rights). Let k be the number of balls drawn from the urn. The probability of successful transmission is that of getting a single black ball in the sample. This probability is given by an element of the Hypergeometric distribution

$$\frac{\binom{k}{1}\binom{N-k}{n-1}}{\binom{N}{n}}$$

The last expression assumes its maximum when $k=\lfloor N/n \rfloor$ ($\lfloor x \rfloor$ gives the integer part of x). Not only does this value of k maximize the probability of selecting exactly one black ball, but it also gives that the average number of black balls selected is equal to unity. (This result is not unlike Abramson's optimality condition [2], $1=G_1+G_2+\ldots+G_N$ for Slotted *ALOHA* (where G_i is the transmission probability of the i-th PRU). Abramson's condition is such that the optimal choice of transmission probabilities for Slotted *ALOHA* gives an expected number of transmitted packets during a slot equal to unity. In fact, both optimality rules are shown[7] to be particular instances of a more general optimality condition for assigning access rights.) We call a scheme which allocates transmission rights to a sample of $k=\lfloor N/n \rfloor$ PRUs, an *URN SCHEME*.

Let us reconsider the example of two PRUs in terms of the urn model. When N=2 and n=1, the number of balls to be sampled is k=2, i.e., both PRUs should have access rights. When N=2 and n=2 then k=1, i.e., only one (any one) PRU should transmit. These numbers conform with the previous intuitive solution.

The urn scheme adapts smoothly to the load on the system. When the the system is lightly loaded, a large number of PRUs get the right to share channel slots. For instance n=1 implies k=N; all PRUs get transmission rights (as in Slotted *ALOHA*), however, only one is going to make use of it. As the load increases (i.e., as n grows), k decreases and sharing is gradually restricted. When the system is heavily loaded, sharing of slots is eliminated; indeed, when n>N/2 then k=1 and the urn scheme converges to Time Division Multiple Access (*TDMA*) which is the best scheme for a heavy load. If the sampling of k is random, the urn scheme converges to random *TDMA*; if the sampling of k is without repetitions from slot to slot (until all balls are sampled) the urn scheme converges to round robin *TDMA*.

Unlike optimally controlled *ALOHA* whose channel usage efficiency is 37%, and which uses the same information as the urn scheme, the channel usage efficiency of the urn scheme is 100%. When the traffic is light, the urn scheme permits some waste of the channel in the form of collisions and empty slots; this is the price of partial information. However, when the traffic increases, collisions are smoothly eliminated and the full channel capacity becomes available for actual service.

It is possible to show that the above properties (adaptivity and perfect efficiency) remain valid in the limit when N grows to infinity. Therefore the urn scheme is superior to optimally controlled *ALOHA* and *TDMA*, independent of the system size N. We now proceed to describe some possible implementations of the scheme and results of performance analysis and simulation.

2. *IMPLEMENTATION OF URN SCHEMES*.

The details of implementation may vary according to the particular environment considered. In what follows we consider a few versions of the urn scheme. Our main concern is to expose some issues that arise on the boundary between theoretical models and practical implementation. We do not give a detailed description of the implementation.

Two major issues arise in the implementation of a distributed adaptive access scheme: How does the decision algorithm acquire the information which it requires? How does the the algorithm obtain coordination of the distributed decisions?

2.1 *Information acquisition mechanisms.*

There are a few schemes to acquire information about the number of busy PRUs. For instance, it is possible to estimate n from the acknowledgment traffic. This requires a filter for a jump process. The problem has been solved in conjunction with the problem of controlled *ALOHA* [8]. The solution is general enough to be easily adapted to urn schemes. However, filtering n from the statistics of the acknowledgments is too slow an information acquisition mechanism; thus we shall not dwell upon filtering schemes to estimate n.

Also, it is possible to implement the urn scheme using the acknowledgment traffic directly [7]. This is a result of a general rule for optimal adaptivity, of which the urn scheme is a particular instance. There is no need to estimate n; it is enough to adjust k and monitor the statistics of the acknowledgment traffic. The scheme becomes quasi-static (in the sense of [9]). Quasi-static implementation (like the one suggested here, or in the previous paragraph) is to be preferred for systems with a limited ability to adapt.

Let us consider a possible scheme for estimating n with high accuracy, at the expense of a negligible overhead. All we need is a binary erasure shared reservation subchannel. The reservation channel may be implemented by means of a reservation mini-slot (whose size, unlike perfect reservation schemes[10,11,12], is independent of the total system size) at the beginning of each data slot. An idle PRU which turns busy sends a standard reservation message of few bits. If two or more PRUs send a reservation in the same minislot, the result is an erasure (i.e., a collision). We assume that the PRUs can detect three events over the reservation subchannel: no new busies, one new busy, or erasure. Figure (2) depicts a typical series of events over the channel.

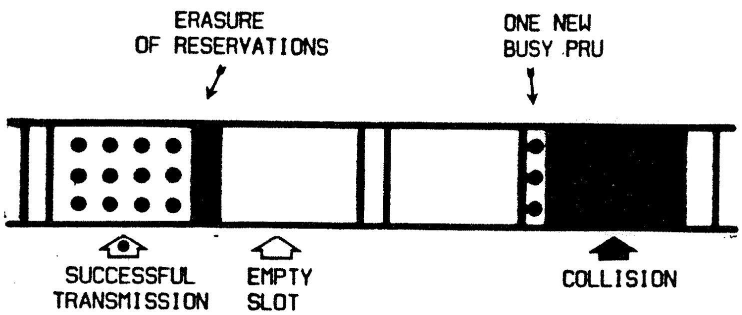


FIGURE (2): A SEQUENCE OF EVENTS OVER THE CHANNEL.

When a busy PRU turns idle the condition is detected by other PRUs from the acknowledgment of his last packet. Therefore the PRUs can follow the number of ***newly idle*** PRUs (with no error) and the number of ***newly busy*** PRUs (with some error due to erasures). Nevertheless, the information acquired by the PRUs is sufficient to determine the number of busy PRUs with a high accuracy. Consider a worst case analysis of the error. Let us consider an infinite system (i.e., $N=\infty$), and let all PRUs be idle, so that each arrival turns an idle PRU busy. Let us also assume that the overall number of new arrivals per slot is given by a Poisson distribution with a rate r ($0 \le r \le 1$ for stability).

The probability of at least two arrivals in the same slot (erasure over the reservation channel) is given by:

$$1-e^{-r}-re^{-r} \le 0.26$$

The probability that more than two arrivals occurred, given an erasure over the reservation channel (i.e., at least two newly busy PRUs), is:

$$1 - r^2/2(e^r-1-r) \le 0.3$$

This conditional error probability is depicted in figure (3) as a function of the input rate r. We note that the probability of more than two arrivals is less than 0.08. Thus estimation errors will occur in at most one out of 13 slots. We see that it is possible to develop a very accurate estimate of n which need only be corrected from time to time. For instance, every time the system goes idle, or every few hundred slots.

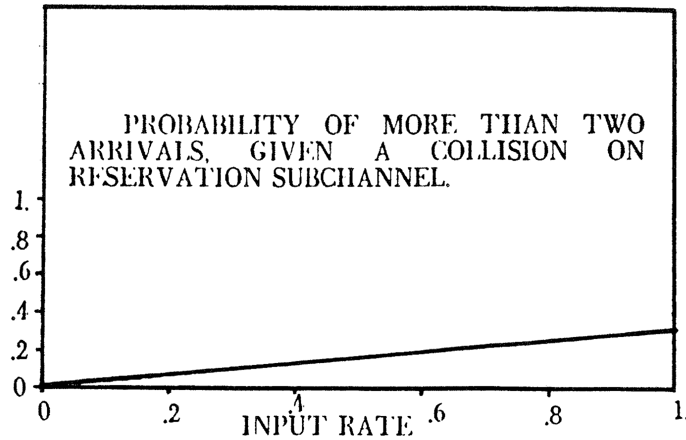


FIGURE (3): CONDITIONAL ERROR PROBABILITY.

Figure (4) illustrates an atypical sample path of the estimated n and its actual value, during a very long busy period, , as obtained from simulation. (Typical busy periods were shorter and the estimated n almost never diverged from the actual n; thus we had to use an atypical sample path to display the divergence of the estimate.) The estimation strategy which was simulated is to keep an underestimate of n i.e., an erasure is counted as two arrivals. Errors in estimation are detected when the estimated n becomes 1 and all PRUs have transmission rights. If the actual value of n is greater than 1, a collision occurs (indicating that our estimate n=1 was in error and was too small) and the estimated n is increased by one. The sample path in figure (4) was generated by a system with 10 PRUs for a total arrival rate of r=0.8 packets per slot. For lower traffic, simulated errors were very rare.

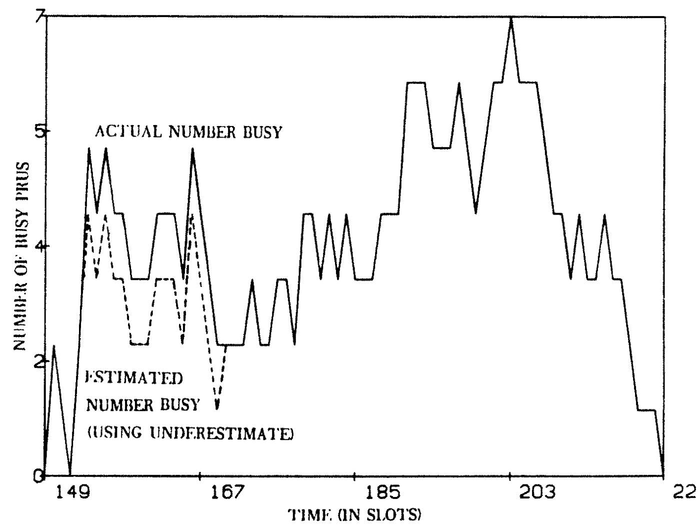


FIGURE (4): ERROR IN UNDERESTIMATION.

2.2 *Coordinated decision.*

The urn scheme requires that the PRUs agree upon k, the number of PRUs to be selected for access rights, and their identity. It is important to note that the scheme is robust w.r.t. errors in the two decisions. That is, the optimal $k=\lfloor N/n \rfloor$ is insensitive to small perturbations of n. Errors in k or failure to agree upon the identity of the k lucky PRUs do not lead to a complete breakdown. It is possible to build effective measures to restore normal operation every few hundred slots, to account for possible malfunctioning.

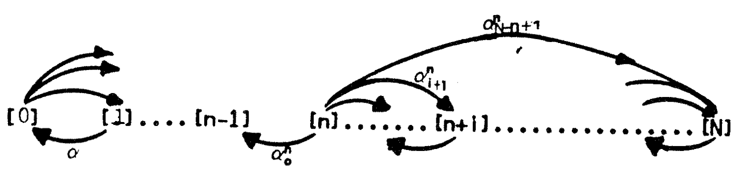
The selection of which k PRUs should get access rights may be obtained through a preprogrammed priority mechanism. For example, it is possible to employ a pseudo random number generator which uses the same seed at each PRU. Each PRU draws the same k numbers between 1 and N. Thus the lucky PRUs know who they are and know that they may transmit. This we shall call the random (or basic) urn scheme, to distinguish it from other implementations. The random urn scheme may be improved if we draw a permutation of the numbers from 1 to N and take k PRUs at a time, during each slot, avoiding repetitions during a cycle over the whole system. (The improvement is similar to that obtained by a round-robin *TDMA* over random *TDMA*).

An even more effective mechanism can make use of the asymmetric information which is acquired from the asymmetric sampling policy. We use a round-robin slot sharing *WINDOW* mechanism. The PRUs are ordered according to their numbers along an imaginary circle. The k lucky PRUs are selected by a window which rotates around the circle. When a collision occurs, the window stops and decreases in size (say to half its previous size); the process is repeated each slot until a success occurs. If there was no collision, the tail of the window is advanced along the circle to the head of the previous window and the window size is once again set equal to k (based on the current estimate of n). Window schemes offer an improvement over the basic urn scheme. The selection of PRUs to be given transmission rights, adapts to both the total load and the results of the selection of the previous slot. It is possible to design window schemes with memory span longer than one slot. Such algorithms may offer some improvement of the performance at the expense of higher complexity and lower reliability. We do not pursue that subject here.

3. *ANALYSIS AND SIMULATION.*

Access schemes are essentially service mechanisms of a queueing process. To analyze the performance of a scheme it is required that we model the arrival process and describe the queueing mechanism. We compare the performance of the basic (random) urn scheme to that of: optimally controlled Slotted *ALOHA* (i.e., with transmission probability 1/n); random *TDMA* (we use random *TDMA* to avoid the non-stationary behavior of the service mechanism from slot to slot); and perfect scheduling of the channel.

Arrivals are assumed to be independent from slot to slot and drawn from a time independent distribution. The service mechanism for the four different schemes (urn, optimal *ALOHA*, *TDMA* and perfect scheduling) is independent of the arrival process or the time. (The service depends, however, upon the number of busy PRUs). We assume that each PRU possesses a single buffer. Thus the number of busy PRUs is a finite Markov chain whose transition structure is depicted below



TRANSITION DIAGRAM FOR THE "NUMBER BUSY" PROCESS.

The transition matrix of the chain is given by an upper Hessenberg matrix. The steady state equation for the distribution of the number of busy PRUs has the form

$$(\pi_0,\pi_1,\ldots\pi_N)=(\pi_0,\pi_1,\ldots\pi_N)\begin{bmatrix}\alpha_1^0 & \alpha_2^0 & \cdots & \alpha_{N+1}^0\\ \alpha_0^1 & \alpha_1^1 & \cdots & \alpha_N^1 \\ 0 & \alpha_0^2 & \cdots & \\ & 0 & & \vdots \\ & & 0 \quad \alpha_0^N & \alpha_1^N\end{bmatrix}$$

Here π_n is the steady state probability of finding n busy PRUs in the system.

The steady state equations can be solved by a simple recursive routine. The solution may be used to compute the expected throughput and delay (using Little's result [3]). We applied this analytical method to the four different schemes. Figure (5) depicts typical delay-throughput curves for a system with N=10 PRUs. Figure (6) depicts the throughput as a function of the input rate for the four schemes. It may be seen that the random urn scheme adapts smoothly to the load; it is equivalent to optimally controlled Slotted *ALOHA* when the input load is light, but converges to *TDMA* when the load increases. In the medium range it is better than both. The difference between the performance of the urn scheme and perfect scheduling represents the price of imperfect information.

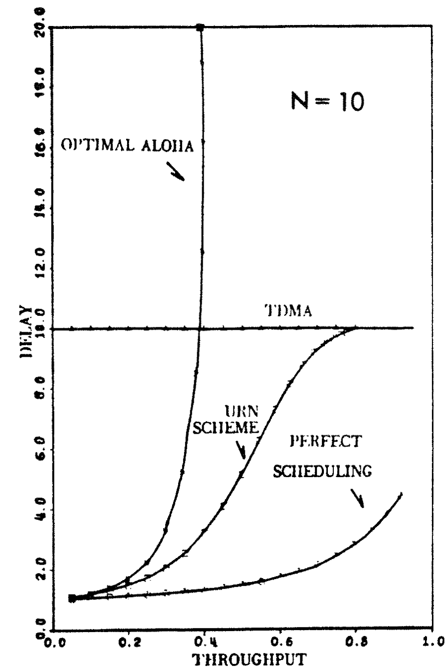


FIGURE (5): DELAY-THROUGHPUT COMPARATIVE ANALYSIS OF THE FOUR SCHEMES.

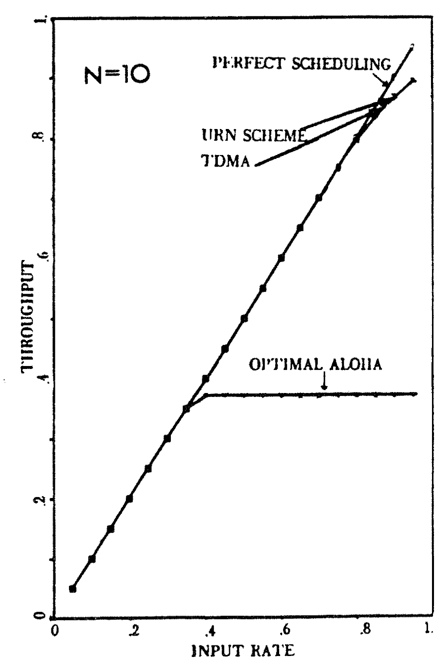
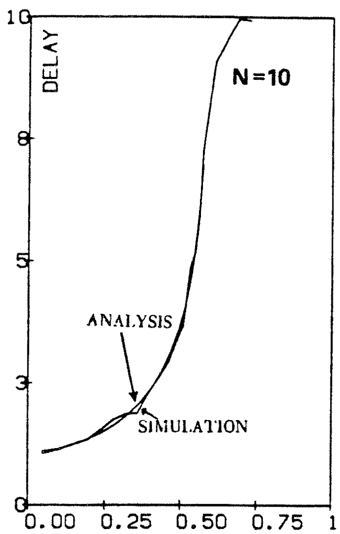


FIGURE (6): INPUT-THROUGHPUT ANALYSIS OF THE FOUR SCHEMES.

The random urn scheme and the window scheme were also simulated and compared with optimally controlled Slotted *ALOHA*, *TDMA* and perfect scheduling. Figures (7) and (8) compare the results of simulation and analysis. Figure (9) show the performance of the four schemes when each PRU has a buffer for 25 packets. Finally, figure (10) compares the delay-throughput performance of the random urn scheme and the window scheme of the previous section. It may be seen that the window scheme improves the performance significantly.



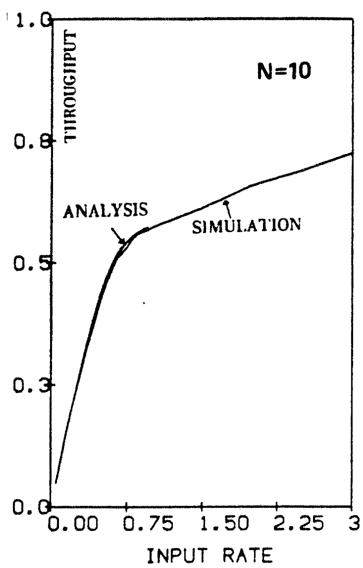


FIGURE (7): DELAY-THROUGHPUT FIGURE (8): INPUT-THROUGHPUT

ANALYSIS AND SIMULATION OF THE RANDOM URN SCHEME

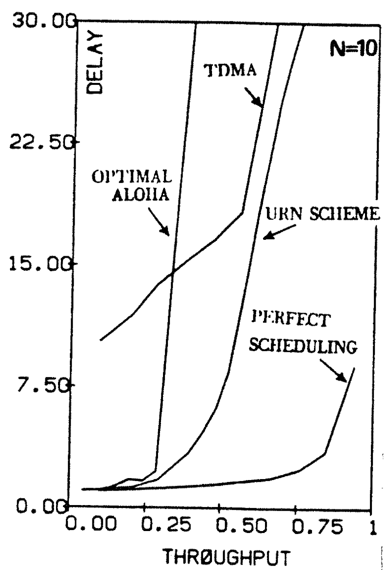


FIGURE (9): DELAY-THROUGHPUT OF THE FOUR SCHEMES. SIMULATION FOR BUFFERED PRUS.

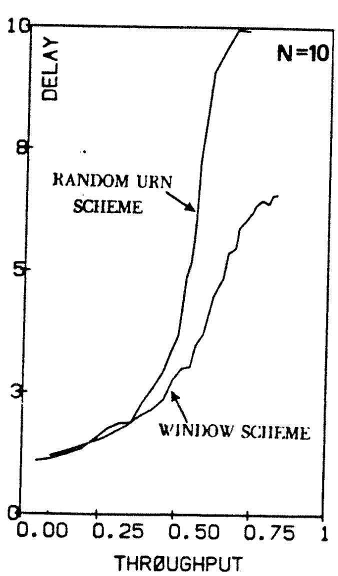


FIGURE (10): DELAY-THROUGHPUT OF THE SIMULATED RANDOM URN AND WINDOW SCHEME.

The question of robustness is currently under study. Also, urn schemes may be employed in a multi-hop environment [7]. The idea is simple. Each PRU may reside in two or more urns simultaneously. Access rights need to be resolved in all of those environments at the same time. It is possible to use a simple frequency division into three bands to localize the interaction between different environments. The network may be decomposed into local environments each of which looks almost similar to a one-hop system, and employs the urn scheme as a method to resolve the conflict over channel usage. These results will be published in a forthcoming paper.

4. *CONCLUSIONS.*

Urn schemes offer an attractive class of adaptive sharing policies for multiple access broadcast channels. The basic urn scheme performs better than optimally controlled Slotted *ALOHA*, which uses the same information for adaptivity. The urn scheme retains the simplicity and robustness of Slotted *ALOHA*; yet it does not limit the amount of available channel capacity. Adaptivity is obtained at the price of a negligible overhead. Versions of the scheme represent a feasible solution to the problem of access schemes in a multi-hop environment. Therefore the schemes offer a significant advancement from the practical point of view.

From a theoretical point of view, urn schemes demonstrate an optimal performance bound on adaptive access schemes which use symmetric state information only. Further research is required to develop a systematic attack on the problem of partial non-homogeneous information and the problem of coordination for real time distributed decision processes.

Acknowledgment:

We wish to thank Johanna Moore, Lou Nelson and Frank Heath for their help in simulating and plotting the results.