IEEE P802.11
Wireless Access Methods and Physical Layer Specifications

Performance of a Reservation Multiple-Access Protocol

Presented by:
Richard O. LaMaire
IBM Research Division
T. J. Watson Research Center
P. O. Box 704, Rm. H2-B04
Yorktown Heights, NY 10598
(914) 784-7571
lamaire@watson.ibm.com

Abstract - The performance of the Medium Access Control (MAC) protocol that was described in [1] is analyzed. The high throughput that can be achieved as a result of using reservations is illustrated. We also describe the performance advantages of using an adaptive procedure for tuning a key protocol parameter based on the estimated number of active stations. Additional features of the reservation MAC protocol are described including its fixed frame size which facilitates the inclusion of an isochronous service.

1 Introduction and Protocol Description

We first briefly review the reservation MAC protocol of [1]. In section 2, we describe the traffic models that will be used to evaluate the performance of the MAC protocol. Several examples that illustrate the throughput efficiency of the protocol are presented in section 3. Our conclusions are provided in section 4.

We motivate the choice of a reservation multi-access protocol by first considering the requirements for a wireless MAC protocol. In a wireless system, the access protocol must allow: 1) access by newly activated (or newly arrived) remote stations, and 2) efficient bandwidth utilization. To achieve requirement 1, we chose a fixed frame structure with a portion dedicated to random-access using the slotted Aloha protocol. To help achieve requirement 2, the random-access portion is used to make reservations for other portions of the fixed frame. That is, since the maximum throughput of the slotted Aloha protocol is only about 37%, we seek to minimize the portion of the frame that is devoted to this less efficient transmission scheme. By using reservations, we use the remainder of the frame (i.e., the non-random access part) with 100% efficiency.

The MAC protocol of [1] uses the fixed frame structure that is shown in Fig. 1. (A related reservation protocol was presented in [2].) With reference to Fig. 1, a frame is composed of three periods: the $\mathcal{A}$ period during which traffic is transmitted from the access point (i.e., the base station) to the remote stations, the $\mathcal{B}$ period during which traffic is received by the access point from the remote stations (or during which peer-to-peer transmissions occur), and the $\mathcal{C}$ period, during which reservations are made by the remote stations for service during the $\mathcal{B}$ period. The lengths of the slots in the data periods, that is, the $\mathcal{A}$ and $\mathcal{B}$ periods is denoted by T_d, while the length of a $\mathcal{C}$ period slot or reservation slot is denoted by T_r time units. The reservation slots are assumed to be small slots (i.e., minislots) relative to the data slots. By using small reservation slots, the fraction of the frame that is devoted to the $\mathcal{C}$ period can be reduced resulting in more efficient bandwidth utilization. We denote the ratio T_d/T_r by β and assume that it is an integer. In a wireless system, the successful transmission of a reservation or data slot is acknowledged by the receiving remote or access point.

Although the frame is fixed, the boundaries between the $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ periods are not fixed, but vary according to the $\mathcal{A}$ and $\mathcal{B}$ period traffic requirements. These movable boundaries allow the protocol to flexibly respond to changing traffic profiles (i.e., different mixtures of $\mathcal{A}$ and $\mathcal{B}$ period traffic). The combined length of the $\mathcal{A}$ and $\mathcal{B}$ periods, that is the data periods, is made as large as is required by the traffic subject to the constraint that it is bounded by the design parameter, d_{max}, which is given in units of $\mathcal{C}$ slots. This constraint guarantees that the $\mathcal{C}$ period never becomes smaller than $c_{min} = f - d_{max}$, where c_{min} and the fixed frame length f are also given in units of $\mathcal{C}$ slots. At the beginning of each frame, the access point informs all of the remote stations, via a broadcast mechanism, of the length of the $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ periods.

The random-access $\mathcal{C}$ period operates using a slotted Aloha type of protocol as described in [3]. In our system, each of a finite number of remote stations attempts to transmit in a given $\mathcal{C}$ slot with the same probability p. The stochastic decision to transmit or not is made independently at each of the remote stations. If at most one remote station attempts to transmit in a slot, then that station has successfully contended for transmission. If more than one remote station attempts transmission in a slot, then they collide and must both try to transmit in a later slot. A collision is detected by a remote station by the lack of a confirming acknowledgment signal.

In our analysis, we consider two approaches for choosing the probability of transmission, p. The first approach is to use a fixed value of p for all time slots. A second approach is to vary the value of p that is used in each slot based on the number of active remote stations that are attempting to transmit. We will refer to the choice of p that maximizes the probability of successful transmission as the optimal value of p. While it is not possible to always determine how many remote stations are contending for a given slot, it is useful to consider the performance of the optimal p case since it is an upper bound on practical schemes for varying p. We have observed near optimal performance results from some simple schemes that we have developed for estimating the number of active users. It can be shown that in the slotted Aloha protocol, the optimal value of p is given by $\frac{1}{K}$, where K is the number of active remote stations.

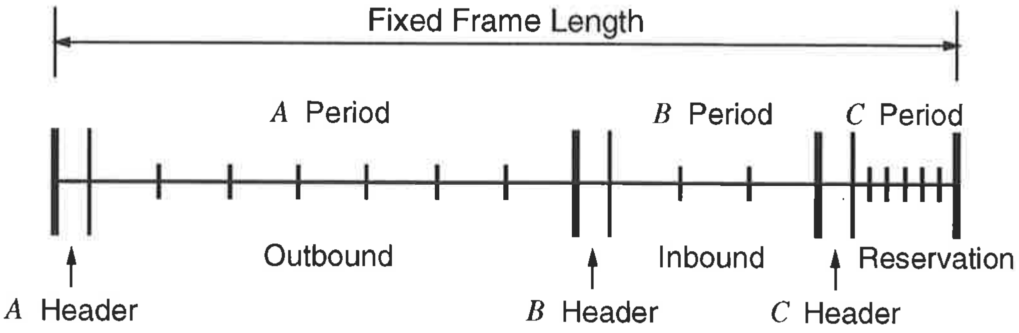

Figure 1: Frame Structure.

2 Traffic Models

The basic traffic model that we consider is shown in Fig. 2. As was described previously, in the $\mathcal{C}$ period, remote stations make reservations in order to obtain reserved transmission time during the $\mathcal{B}$ period so that a remote can transmit either to the access point or to another remote station. In the following discussion, we will refer to this procedure of receiving a reservation and transmitting in the $\mathcal{B}$ period as receiving a *response*. In addition, we will refer to the reservations as *requests* in the following discussion. Although, the reservations that are made during the $\mathcal{C}$ period are made only for the $\mathcal{B}$ period, a remote station can also send a short request for data to the access point during the $\mathcal{C}$ period. In our model, we assume that the data requests that are received by the access point are served in a first-come first-served manner.

In summary, the *requests* in our model take two forms: 1) requests for data to be sent to the remote station from the access point during the $\mathcal{A}$ period, and 2) requests for reserved transmission time during the $\mathcal{B}$ period. Further, the corresponding *responses* take the following forms: 1) a data response from the access point that is transmitted to the remote during the $\mathcal{A}$ period, and 2) the receiving and using of reserved transmission time during the $\mathcal{B}$ period. Using this nomenclature, we describe different operational models for these request and response pairs.

In Fig. 2, a remote station sends a request that is one $\mathcal{C}$ slot in length. Depending upon the type of request, the access point responds by providing either a data response of α slots in the $\mathcal{A}$ period, or by granting a reservation of α slots which is then used by the remote station for transmission in the $\mathcal{B}$ period. We consider two types of traffic models, an open-loop and a closed-loop one, that differ in their policies for determining when the remote station is permitted to generate a new request.

2.1 Open-Loop Peer-to-Peer Model

In the open-loop case, a remote station generates requests independent of whether or not it has received a response from the access point. That is, all K remote stations are always trying to transmit in a $\mathcal{C}$ slot. In this case, a single remote station can have many outstanding requests that are stored at the access point. This open-loop model can be used to represent a peer-to-peer (i.e., remote-to-remote) data transfer in which case the response would be in the form of the completed data transmission corresponding to time that is reserved and used in the $\mathcal{B}$ period. Alternatively, the open-loop model could represent a client-server traffic environment in which additional requests can be made by the remote station client without waiting for the $\mathcal{A}$ period data response corresponding to an outstanding request.

2.2 Closed-Loop Client-Server Model

In the closed-loop traffic model, the remote station may not generate a new request until it has completely received its response (e.g., a data response from the access point). In the closed-loop case there can never be more than one outstanding request per remote station. Since, for this case, a remote station is effectively disabled when it is awaiting a response, there can be less than K remote stations that are trying to transmit in a $\mathcal{C}$ slot. The closed-loop traffic model is useful for representing client-server traffic environments in which the remote station waits for a data response before generating its next request.

2.3 Response Length Distributions

We will consider two types of distributions for the response lengths: 1) a constant length, and 2) a geometrically-distributed length. The geometric response length case can be used to model traffic loads in which the responses have greatly varying lengths. In both cases, our basic unit of time will be the length of a $\mathcal{C}$ period slot. We note that if the mean response length (i.e., the mean of α in Fig. 2) is $\bar{\alpha}$ $\mathcal{A}$ or $\mathcal{B}$ slots, then the mean response length in terms of $\mathcal{C}$ slots is given by

$$m = \bar{\alpha}\,\beta. \tag{1}$$

3 Performance Examples

In this part of the contribution, we show numerical results for several examples that we motivate through some specific assumptions of medium speed and frame size. These results were computed using the Markov chain analysis that is described in [4].

3.1 Motivation

The numerical results that are presented for the following examples were computed in terms of slot times independent of the particular medium speed. However, to facilitate discussion, we will choose a specific medium speed so that the waiting times and transfer delays can be discussed in terms of milliseconds rather than just slot times.

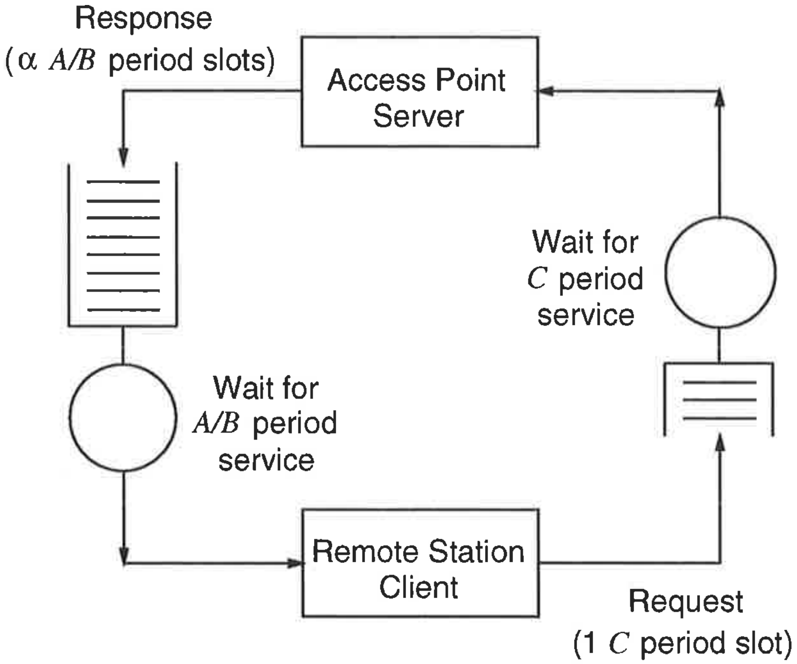

Figure 2: Traffic Model.

We assume a medium speed of 2 Mb/s and a frame length of 5 ms in which the $\mathcal{A}$ and $\mathcal{B}$ slots are each 0.5 ms in duration and the $\mathcal{C}$ slots are 0.1 ms in duration. In this case, a data slot (i.e., an $\mathcal{A}$ or $\mathcal{B}$ slot) is 1000 bits (125 bytes) long and a reservation slot is 200 bits (25 bytes) long. For the client-server model, we assume that each request requires a response of 2 data slots (i.e., 250 bytes) which corresponds to an m value of 10 $\mathcal{C}$ slots. Thus, our client-server model could be used to describe a file server that provides 250 byte responses.

3.2 Open-Loop Peer-to-Peer Examples

We consider several open-loop examples that have constant length responses. In Fig. 3, we show the mean total throughput for several fixed values of p and for the optimal p. In the open-loop case, the optimal p value is $\frac{1}{K}$ for each different number of remote stations value K. By mean total throughput, we refer to the total throughput of all three periods, that is, both the $\mathcal{A}$ and $\mathcal{B}$ data periods and the $\mathcal{C}$ reservation period. From this figure, it can be seen that the use of the optimal p results in good total throughput over the entire range of K, the number of remote stations, whereas the use of a fixed p only yields good throughput for ranges of K that are close to $\frac{1}{p}$. For Fig. 3, we used $c_{min} = 5$ and chose the access point buffer size for the data periods (i.e., the $\mathcal{A}$ and $\mathcal{B}$ periods) to be large enough to yield negligibly small values of blocking probability for all values of K. It is worth noting that the curves of Fig. 3 do not change if the responses are geometrically distributed with the same mean response length of 10. This is due to the fact that in the absence of blocking, the mean total throughput of the open-loop system depends only on m, K, and p and not on the response length distribution or c_{min}. It can be shown [4] that if no blocking occurs, the optimal p value is used, and c_{min} is chosen to be small, then for large K (i.e., $K > 10$), the mean total throughput is given approximately by

$$\gamma = \frac{m+1}{m+e}, \tag{2}$$

since for large K, the throughput of the $\mathcal{C}$ period alone is given approximately by e^{-1}. Thus, for $m = 10$, we compute the mean total throughput to be 0.87 for large K, which is consistent with Fig. 3.

In Fig. 4, we show the mean waiting time (not including the transmission time), w_r, of a $\mathcal{C}$ period request for the open-loop constant length example. As was the case for the mean total throughput, when no blocking occurs, the optimal p value is used and c_{min} is chosen to be small, we can derive a simple expression for w_r [4],

$$w_r = K\,(m+e) - 1, \tag{3}$$

in units of $\mathcal{C}$ slots.

In Fig. 5, we show the mean waiting time (not including the transmission time), w_d, of an $\mathcal{A}/\mathcal{B}$ period data response for the open-loop constant length example. This figure is roughly a scaled version of the throughput figure, Fig. 3. The mean transfer time (i.e., the request and response waiting times plus the request and response transmission times) is shown in Fig. 6. This figure shows the end-to-end delay that would be encountered when the given number of remote stations are transmitting in the $\mathcal{C}$ period with the indicated probabilities of transmission p. Thus, the curve for the optimal p value shows the delay for the case when the remote stations transmit at the throughput maximizing rate.

Summary:

- Both high throughput and low delay are observed for peer-to-peer traffic.
- The primary component of the delay is the request (i.e., the reservation) waiting time.

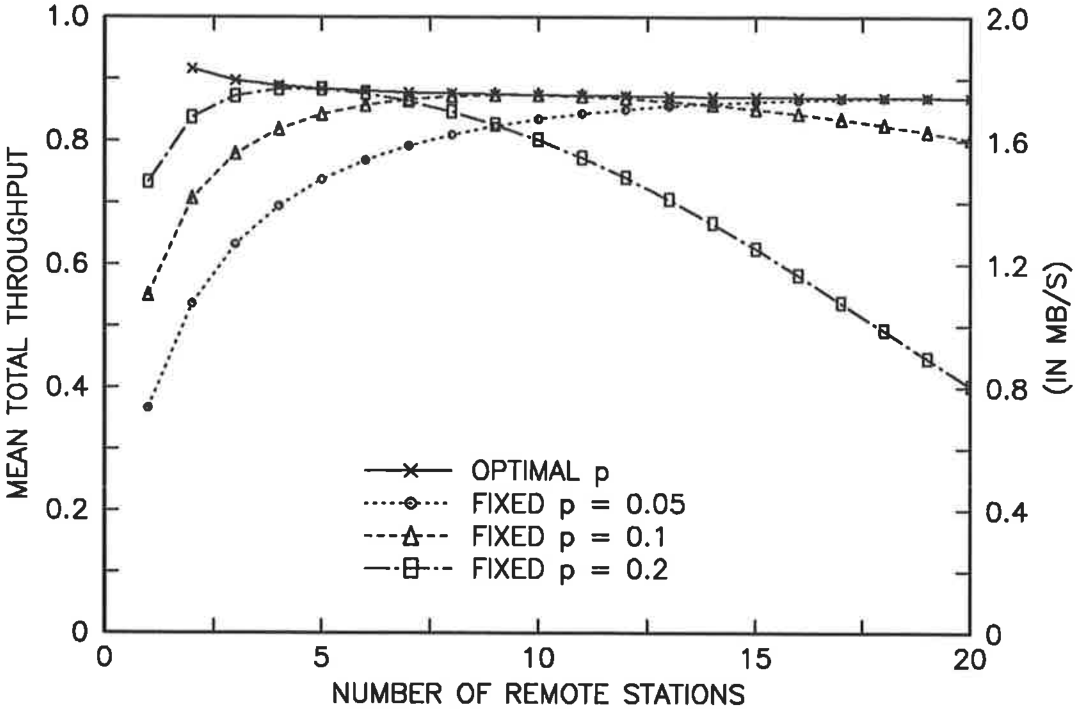

Figure 3: Throughput for open-loop peer-to-peer case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

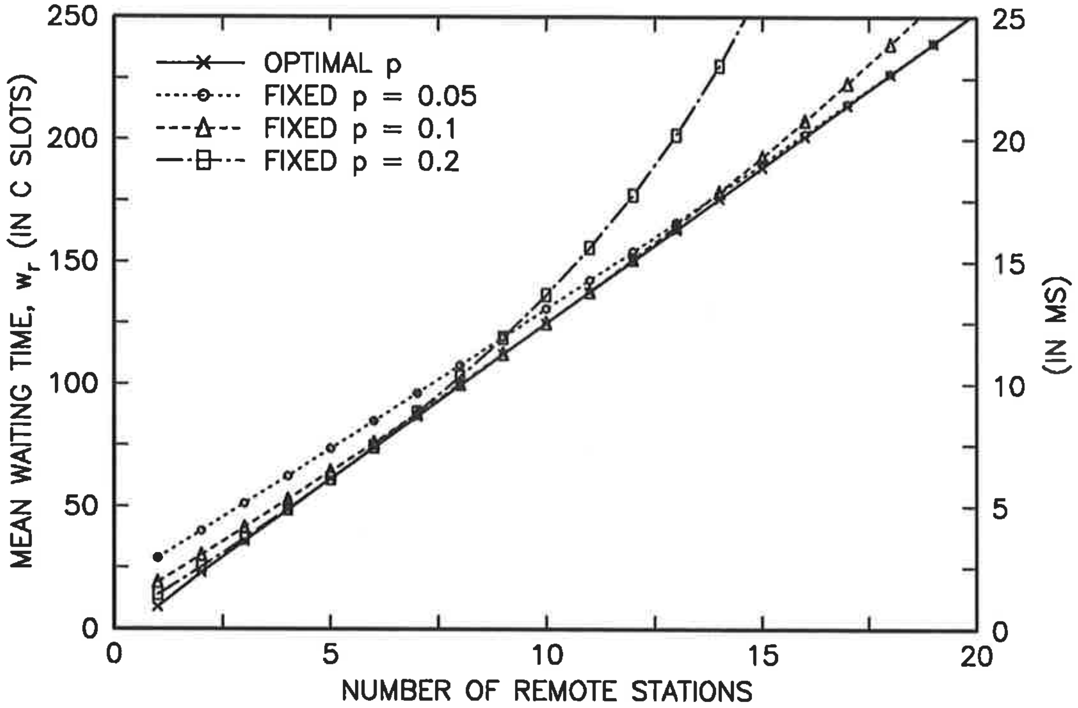

Figure 4: Request waiting time for open-loop peer-to-peer case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

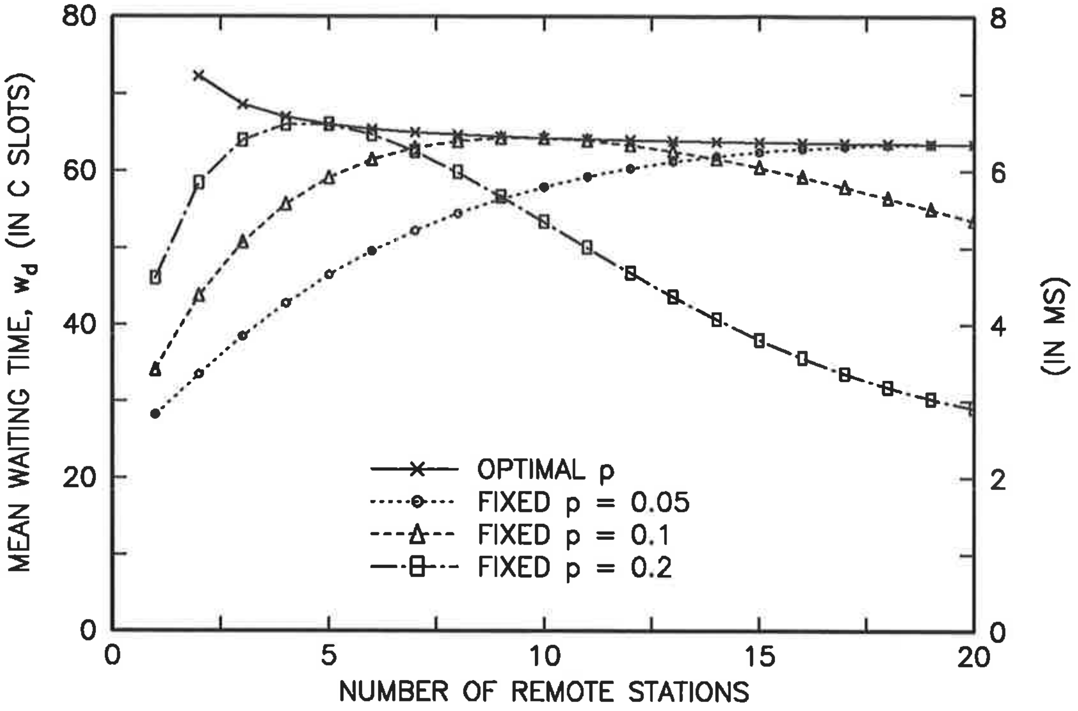

Figure 5: Data waiting time for open-loop peer-to-peer case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

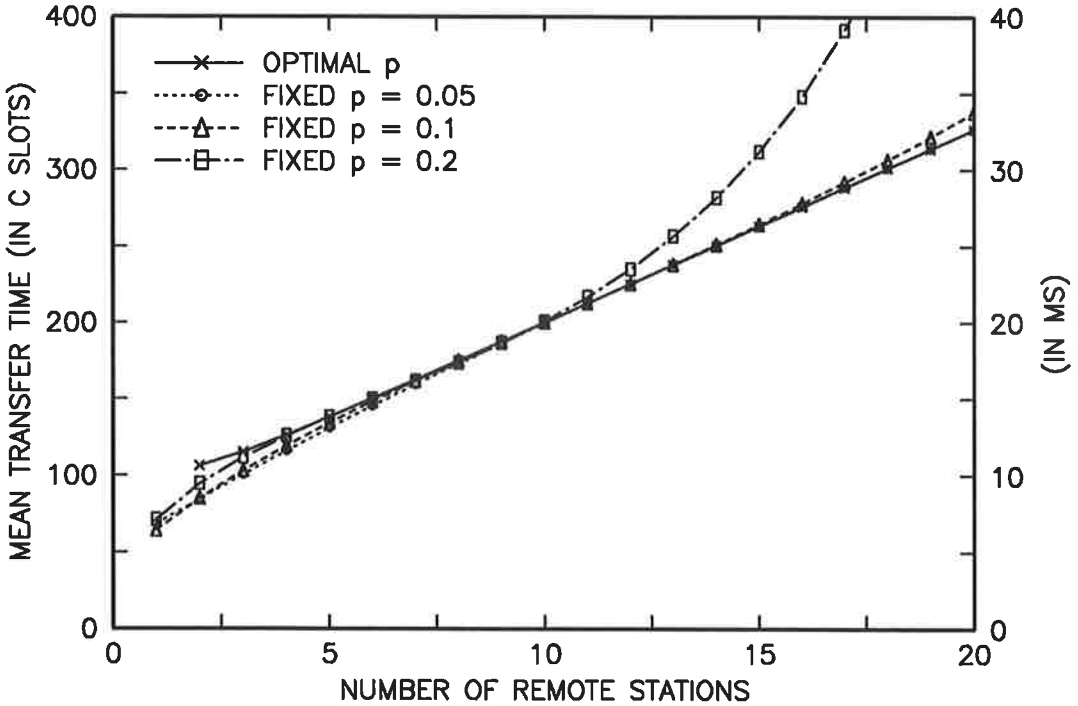

Figure 6: Transfer time for open-loop peer-to-peer case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

3.3 Baseline Closed-Loop Client-Server Examples

We now consider several closed-loop examples that have constant length responses. In Fig. 7, we show the mean total throughput for several fixed values of p and for the optimal p case. It can be seen that the optimal p case yields good total throughput over the entire range of K whereas the use of a fixed p only yields good throughput for certain ranges of K. We note that using a nominal p value of 0.1 does yield relatively good throughput over most of the 0 to 20 range for K. In Fig. 7, we used $c_{min} = 5$ and assumed that the size of the access point buffer for the data periods was larger than K so that no blocking occurs. The mean total throughput for the closed-loop case is also given approximately by Eqn. 2 when K is large, provided that no blocking occurs, the optimal p value is used and c_{min} is chosen to be small.

In Figs. 8 and 9, we show the mean request waiting time, w_r, and the mean data waiting time, w_d, respectively for the closed-loop constant response length example. Further, in Fig. 10, we show the mean transfer time. Note that the mean transfer time corresponds to the expected cycle time for a request/response pair since new requests are not generated by a remote station until the complete response is received. For a small number of remote stations (i.e., < 5), the transfer time is approximately equal to the frame time when the optimal p value is used. Note that for the optimal p, as the number of remote stations increases beyond 5, the mean throughput is approximately constant (see Fig. 7) and the mean transfer time increases approximately linearly.

Summary:

- Both high throughput and low delay are observed for client-server traffic.
- A fixed p of 0.1 performs well for 0 to 20 remote stations.
- An optimal p performs better and can cover a larger number of remote stations.

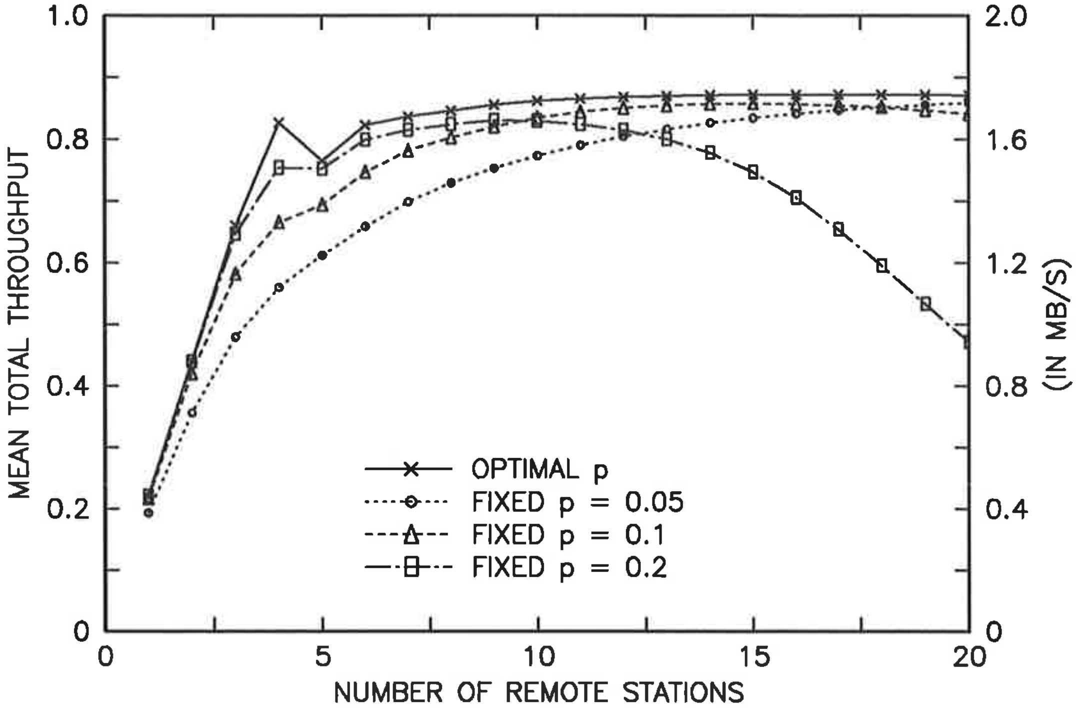

Figure 7: Throughput for closed-loop client-server case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

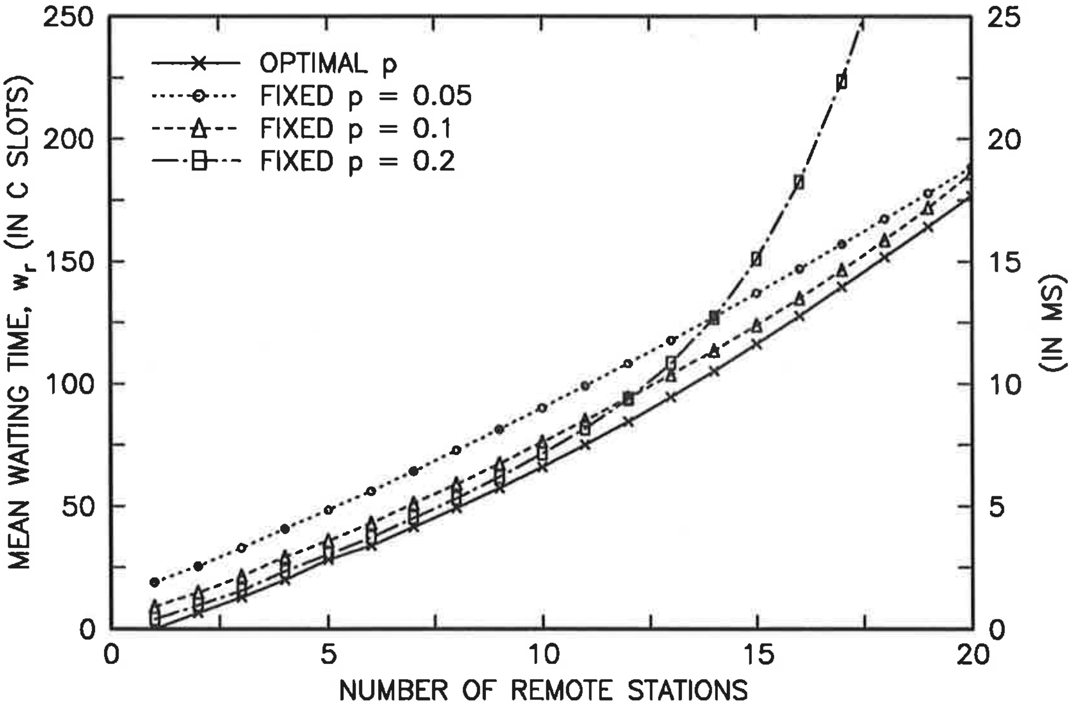

Figure 8: Request waiting time for closed-loop client-server case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

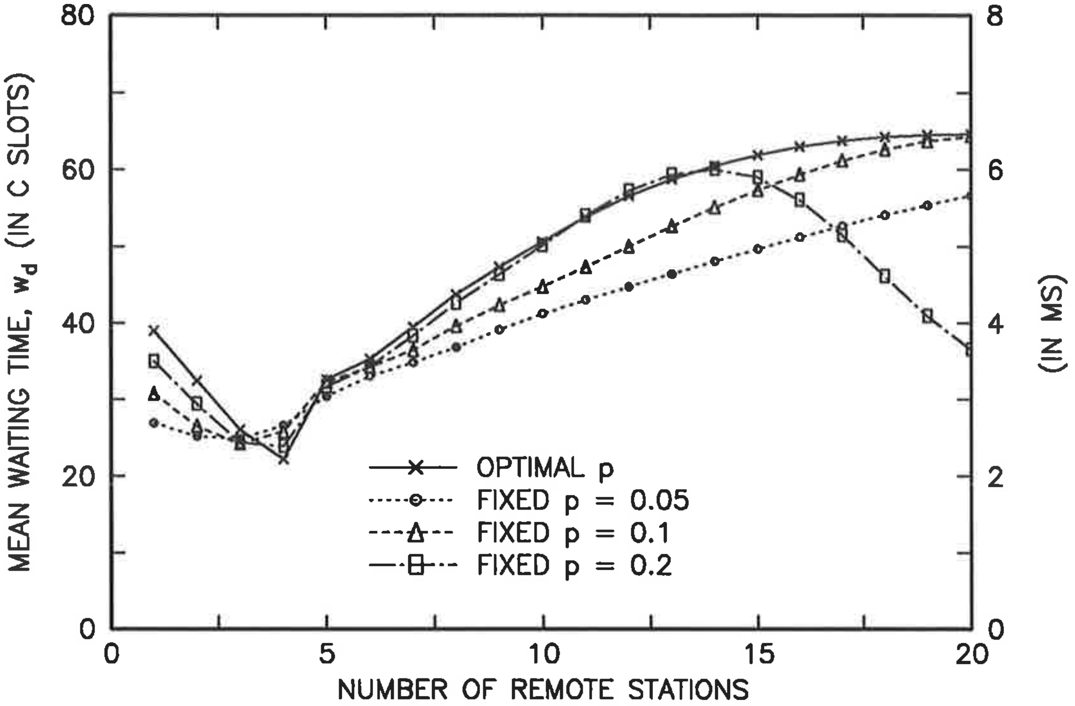

Figure 9: Data waiting time for closed-loop client-server case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

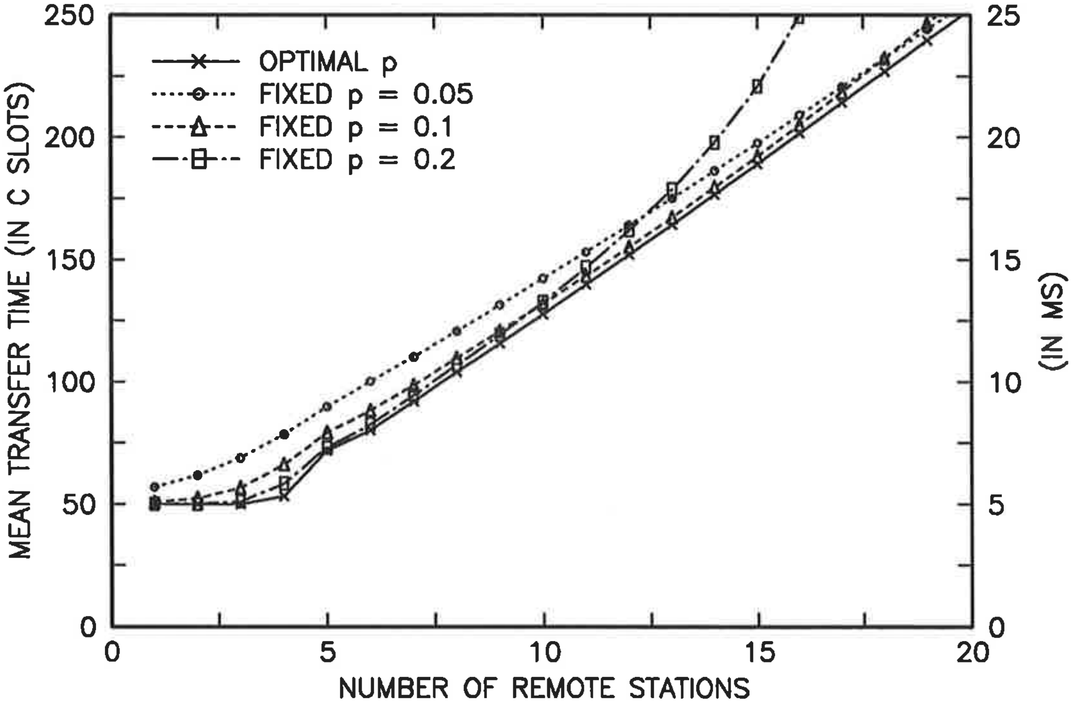

Figure 10: Transfer time for closed-loop client-server case with constant length responses, $m = 10$, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

3.4 Closed-Loop Client-Server Examples for Different Response Lengths

In our next set of closed-loop examples, we examine several cases with different mean response lengths, m. We use the same medium speed, frame length, slot sizes, and c_{min} value as was used in the previous baseline examples. For the transmission probability p, we only consider the case in which the optimal p value is used. We examine cases in which m is 5, 10, and 20 $\mathcal{C}$ slots corresponding to response lengths of 125, 250, and 500 bytes, respectively.

In Figs. 11 and 12, we show the mean total throughput and transfer time, respectively, for constant length responses. The mean total throughput for a large number of remote stations can be found (using Eqn. 2) to be approximately 0.78, 0.87, and 0.92, for the three cases. These values are consistent with the results of Fig. 11. In Fig. 13, we show an alternative representation in which the transfer time is shown as a function of the throughput (or utilization). For low load, the mean transfer time is one frame length. The transfer time does not increase much until the throughput is increased to values that are very near the maximum achievable throughput. Each point in Fig. 13 corresponds to a specific value of K, the number of remote stations. The choice of the optimal p implies that the maximum throughput is being achieved in the reservation period. For a given K value, Fig. 13 shows the highest mean throughput that is achievable and the mean transfer time that is associated with it.

In Figs. 14 and 15, we show the mean total throughput and transfer time, respectively, for geometric length responses. Further, in Fig. 16 we show the corresponding transfer time/throughput curves. These three figures show results that are similar to those of the constant response length case. However, note that the notches of Figs. 11 and 13 (when m is 5 and 10) do not appear in the corresponding figures for the geometric case, Figs. 14 and 16. These notches are caused by some harmonic (or near-harmonic) relationships between the response length m and the maximum combined length of the $\mathcal{A}$ and $\mathcal{B}$ periods, d_{max}. For the constant response length case, such harmonics occasionally lead to some frames that are composed entirely of a $\mathcal{C}$ period in our closed-loop client-server traffic model. In practice, the responses will not all be the same size and there will be some response delays at the server that will reduce the likelihood of a pure $\mathcal{C}$ period frame. The results for the geometric response length case (see Fig. 16) show a much smoother relationship between the mean throughput and the mean transfer time.

Summary:

- Longer responses yield higher throughput efficiency.
- Longer responses yield longer transfer times.
- At low load, the transfer time is approximately one frame length.

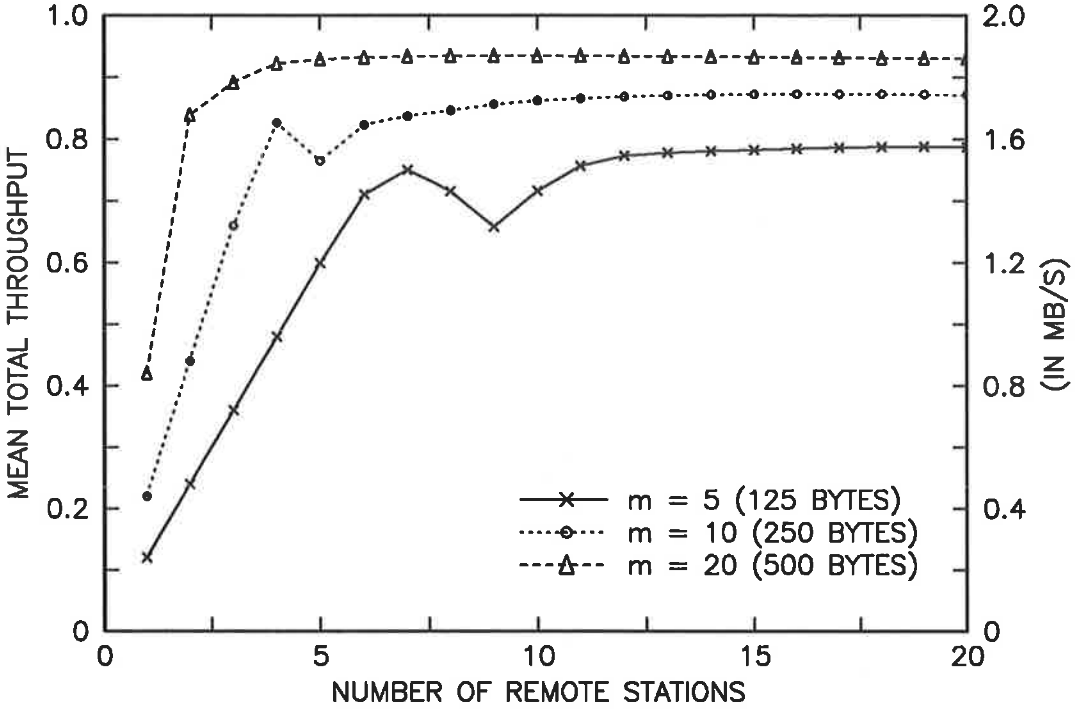

Figure 11: Throughput for closed-loop client-server case with constant length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

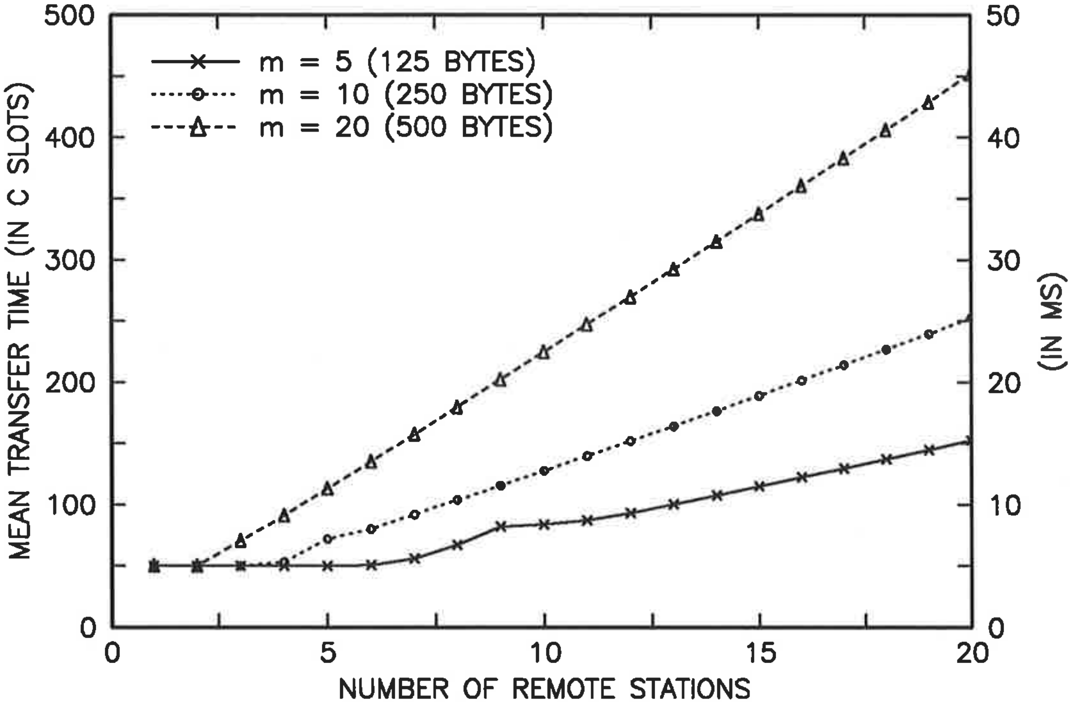

Figure 12: Transfer time for closed-loop client-server case with constant length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

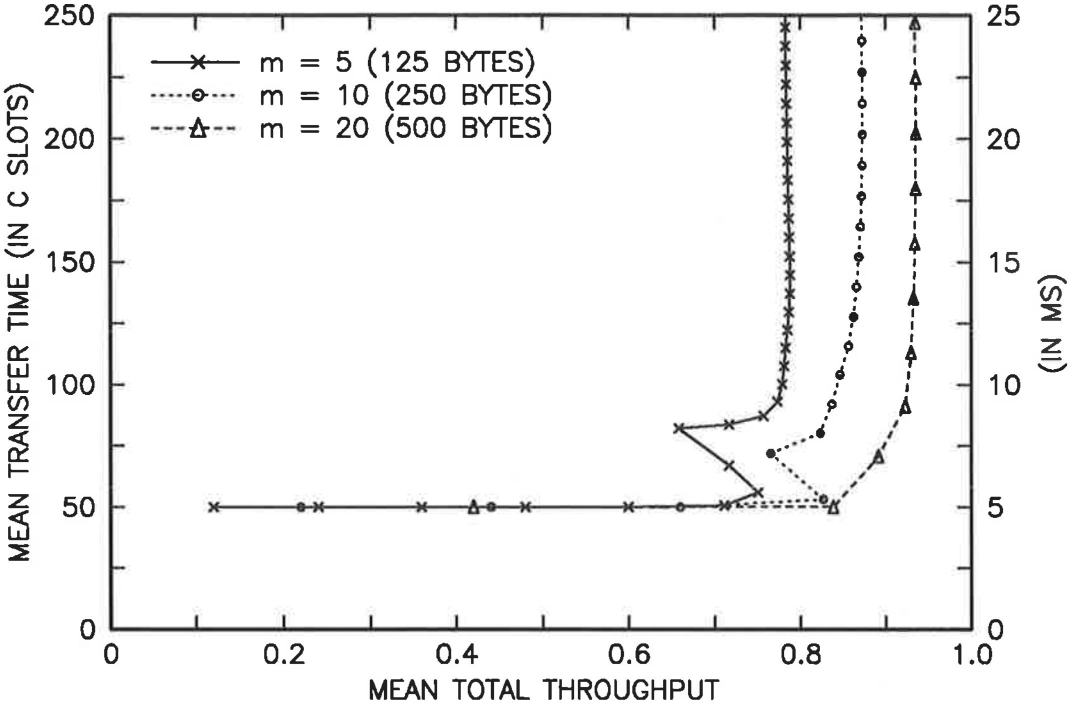

Figure 13: Transfer time/throughput curves for closed-loop client-server case with constant length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

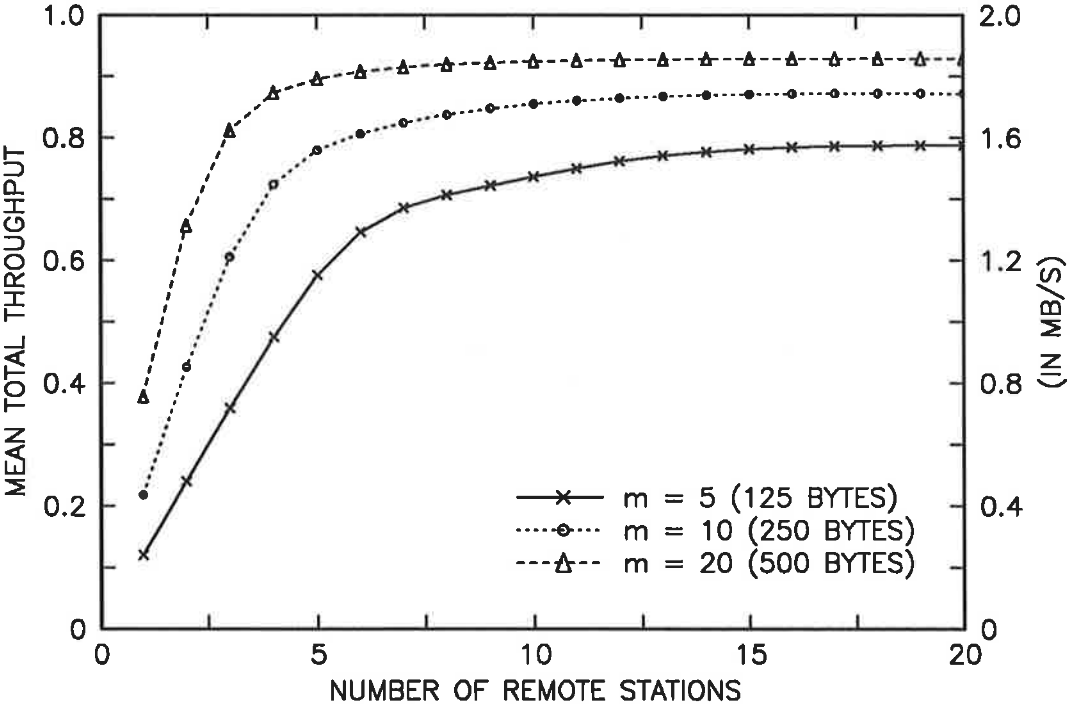

Figure 14: Throughput for closed-loop client-server case with geometric length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

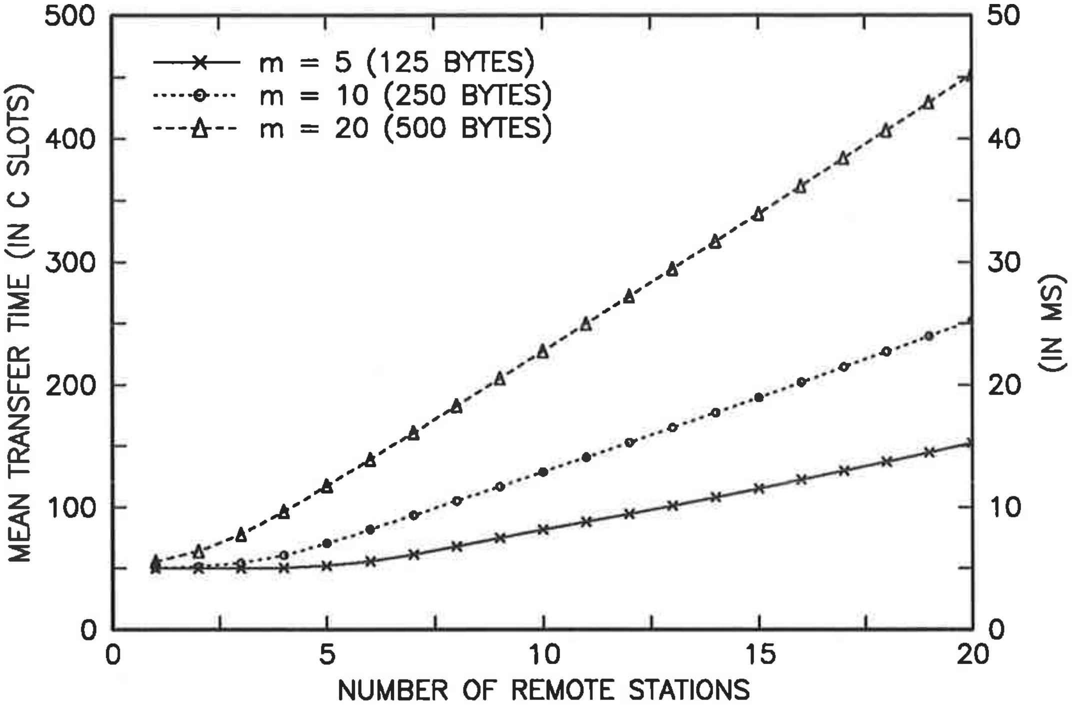

Figure 15: Transfer time for closed-loop client-server case with geometric length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

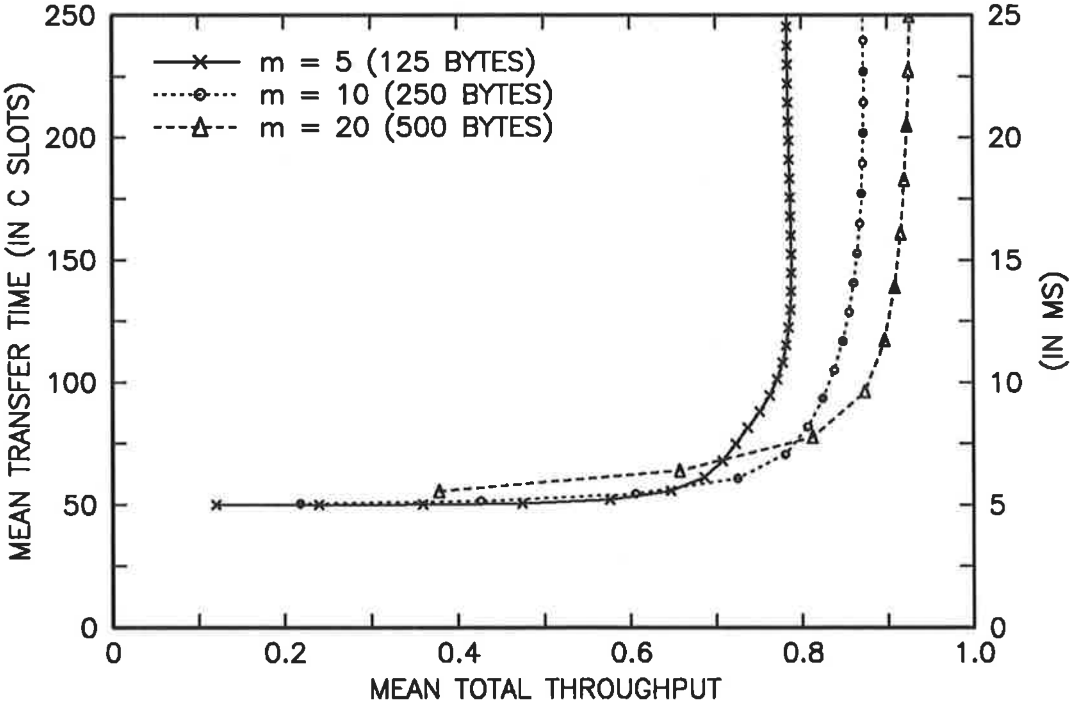

Figure 16: Transfer time/throughput curves for closed-loop client-server case with geometric length responses, optimal p, $c_{min} = 5$, 2 Mb/s medium speed, and 5 ms frame length.

3.5 Closed-Loop Client-Server Examples with Small Reservation Slots

In a second example, we illustrate the advantage of choosing the $\mathcal{C}$ period slots to be small relative to the frame length. In our baseline examples, the frame length was 50 $\mathcal{C}$ slots in length. We now consider a situation in which the frame is 125 $\mathcal{C}$ slots in length. Specifically, we again assume a 2 Mb/s medium speed, but choose the frame length to be 5 ms and a reservation slot (i.e., $\mathcal{C}$ slot) to be 0.04 ms. We assume that the $\mathcal{A}$ and $\mathcal{B}$ slots are 10 times the length of a $\mathcal{C}$ slot and we choose c_{min} to be 10. In this case, a data slot (i.e., an $\mathcal{A}$ or $\mathcal{B}$ slot) is 800 bits (100 bytes) long and a reservation slot is 80 bits (10 bytes) long. For the closed-loop client-server traffic model, we consider cases in which a request requires a response of 1, 2, or 5 data slots. These values correspond to m values of 10, 20, and 50, or responses that have lengths of 100, 200, and 500 bytes, respectively.

Figs. 17 and 18 show the throughput and transfer time results, respectively, for this case with 125 $\mathcal{C}$ slots per frame. As can be seen from Fig. 17, as the response length varies from 20 to 50 $\mathcal{C}$ slots, the mean total throughput varies from 0.92 to 0.94. From Fig. 18, we see that as the response length varies from 10 to 20 $\mathcal{C}$ slots, the mean transfer time is less than 20 ms over the range of remote station populations. Further, at low load, the mean transfer time is equal to approximately one frame length (i.e., 5 ms or 125 $\mathcal{C}$ slots). This example shows that when the $\mathcal{C}$ slots are small relative to the frame length, then the $\mathcal{C}$ period (which has a maximum utilization of approximately 0.37) can be made small relative to the total frame length resulting in high throughput.

Summary:

- The more $\mathcal{C}$ slots that there are per frame, the higher the throughput efficiency.

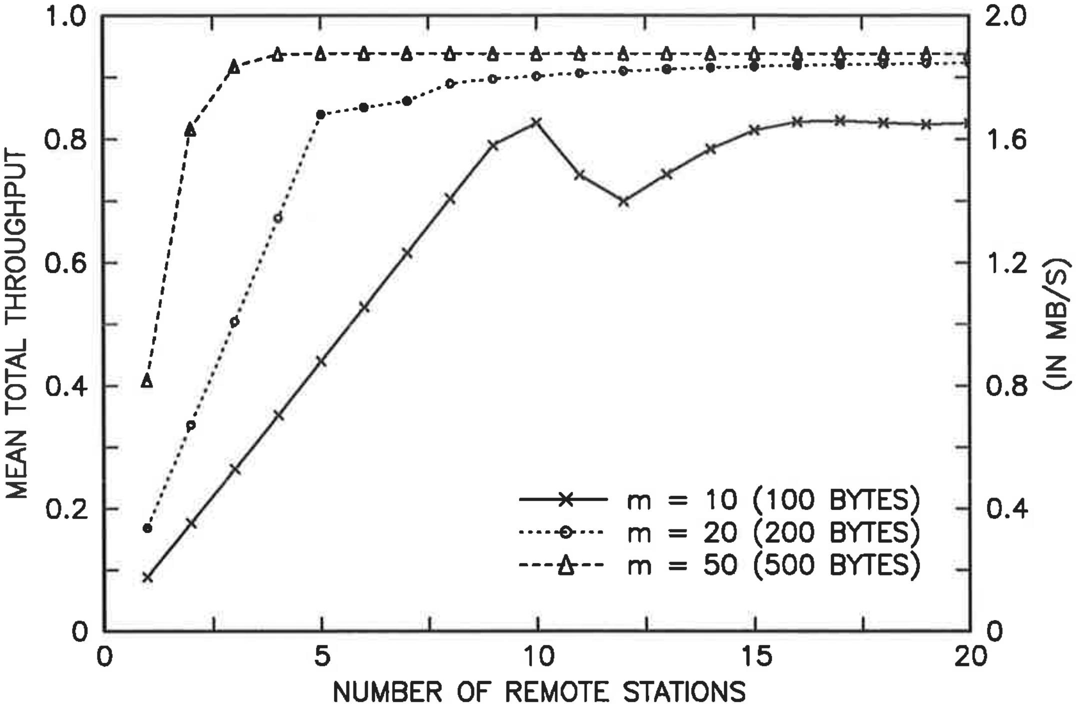

Figure 17: Throughput for closed-loop client-server case with constant length responses, small $\mathcal{C}$ slots, optimal p, $c_{min} = 10$, 2 Mb/s medium speed, and 5 ms frame length.

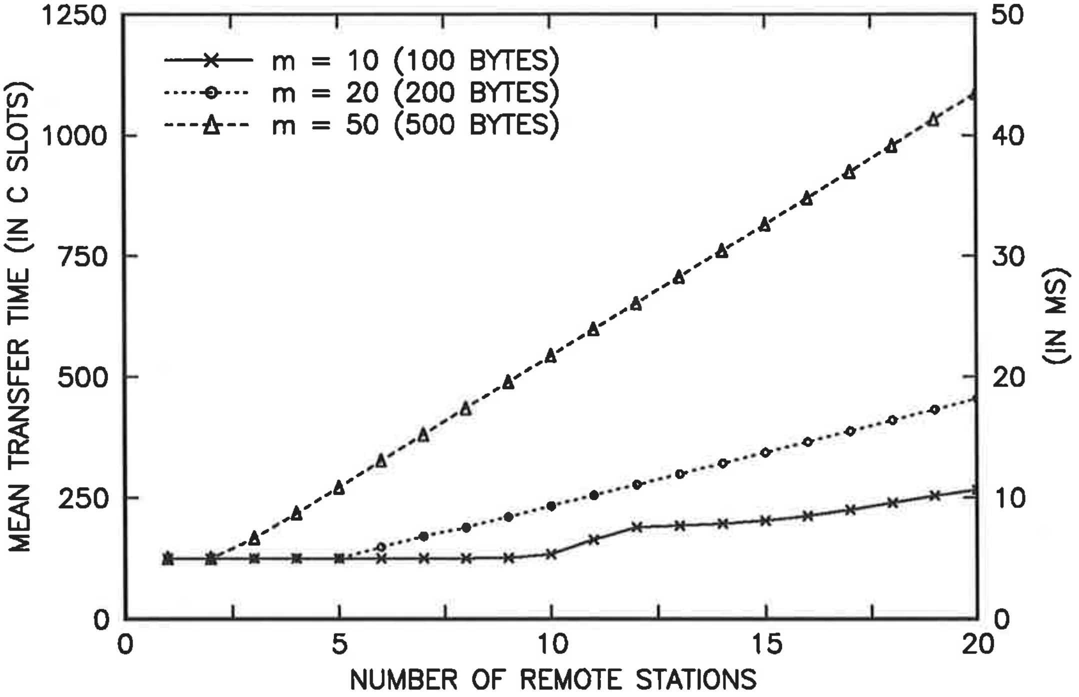

Figure 18: Transfer time for closed-loop client-server case with constant length responses, small $\mathcal{C}$ slots, optimal p, $c_{min} = 10$, 2 Mb/s medium speed, and 5 ms frame length.

3.6 Closed-Loop Client-Server Example with 10 Mb/s Medium Speed

In a final example, we show how the proposed protocol performs at a high medium speed and for up to 40 remote stations. We consider an example with a 10 Mb/s medium speed, a frame length of 1 ms, and a c_{min} value of 5. Further, we assume the use of an optimal p value. As in our baseline set of examples, we use 50 $\mathcal{C}$ slots per frame and a 5 to 1 ratio of the $\mathcal{A}/\mathcal{B}$ slot length to the $\mathcal{C}$ slot length. Thus, the $\mathcal{A}/\mathcal{B}$ slots are 0.1 ms in duration and the $\mathcal{C}$ slots are 0.02 ms is duration. In this case, a data slot (i.e., an $\mathcal{A}$ or $\mathcal{B}$ slot) is 1000 bits (125 bytes) long and a reservation slot is 200 bits (25 bytes) long. In this example, we consider constant length responses with an m value of 10 which corresponds to a response length of 250 bytes. Figs. 19 and 20 show the excellent throughput (i.e., 87% for large K) and transfer time characteristics of the protocol.

Summary:

- The protocol scales well to higher medium speeds and increased populations of remote stations.

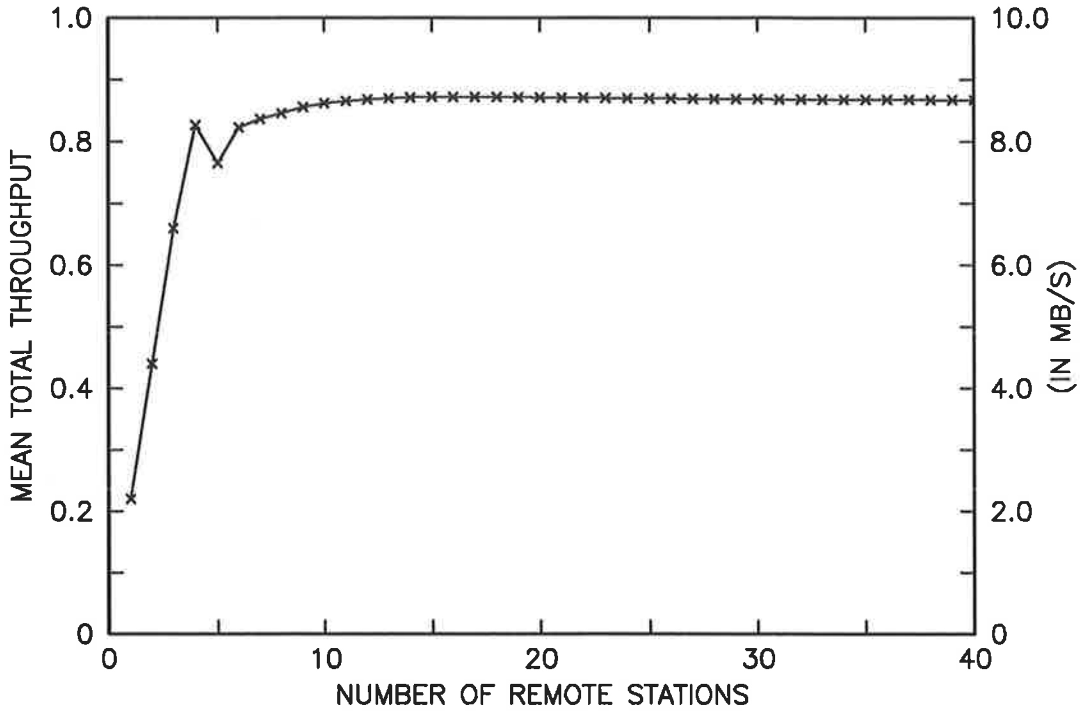

Figure 19: Throughput for closed-loop client-server case with constant length responses, $m = 10$, optimal p, $c_{min} = 5$, 10 Mb/s medium speed, and 1 ms frame length.

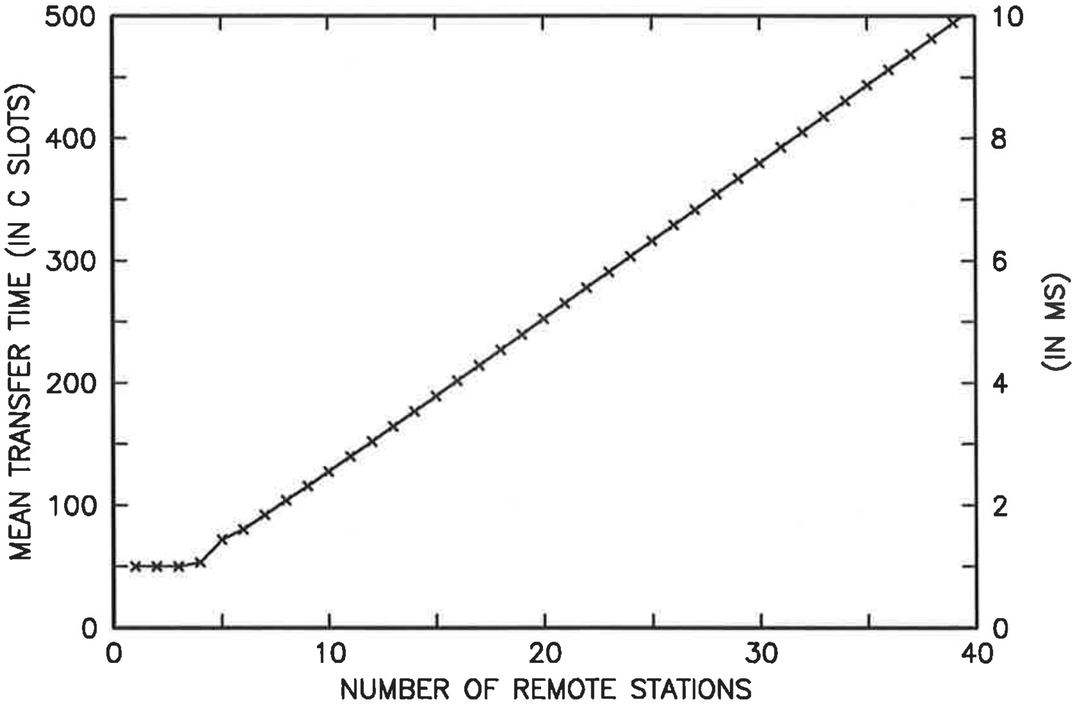

Figure 20: Transfer time for closed-loop client-server case with constant length responses, $m = 10$, optimal p, $c_{min} = 5$, 10 Mb/s medium speed, and 1 ms frame length.

4 Conclusions

In this contribution we have shown that a reservation multiple-access protocol can provide high throughput and good delay characteristics under a variety of traffic conditions. In addition to these good performance results, the proposed reservation multiple-access protocol has the following attributes that make it attractive as a choice for standardization:

1. The proposed MAC protocol flexibly adapts to different mixtures of traffic in the $\mathcal{A}$ and $\mathcal{B}$ periods (i.e., the traffic that is outbound or inbound from the access point).

2. The fixed frame size permits the straight-forward addition of an isochronous service to the protocol. Thus, voice or compressed video can be accommodated along with the current asynchronous data service.

3. The protocol allows for robust performance in the presence of channel errors. In the proposed protocol, response messages are segmented into smaller packets (i.e., packets that are the length of an $\mathcal{A}$ or $\mathcal{B}$ slot.). When used along with a *Go-Back-N* or a *Selective Repeat* error control protocol, the impact of channel errors can be reduced as compared with schemes that do not use message segmentation. The point here is that message segmentation is an inherent part of the proposed protocol. In some other protocols this is not the case.

4. As was shown in the last part of the performance results, the proposed protocol can scale well to higher speeds in which a larger number of remote stations may be contending for the channel.

References

[1] K. S. Natarajan, "Medium access control protocol for wireless lans (an update)," IEEE P802.11/92-39, March 1992.

[2] Y. Takiyasu, "High-performance access control method for base station-controlled systems," IEEE P802.11/92-71, July 1992.

[3] D. Bertsekas and R. Gallager, *Data Networks.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[4] R. O. LaMaire, A. Krishna, and H. Ahmadi, "Analysis of a wireless MAC protocol with client-server traffic," submitted to INFOCOM '93, July 1992.