Computer Networking Symposium 1983

A symposium sponsored by

IEEE Computer Society
Technical Committee on
Computer Communications

U.S. Department of Commerce
National Bureau of Standards
Institute for Computer Sciences and Technology

December 13, 1983
Sheraton-Silver Spring Hotel
Silver Spring, Maryland

ISBN 0-8186-0512-X
IEEE CATALOG NUMBER 83CH1981-0
LIBRARY OF CONGRESS NUMBER 83-82834
IEEE COMPUTER SOCIETY ORDER NUMBER 512

PROCEEDINGS

computer networking symposium

COMPUTER SOCIETY PRESS

IEEE COMPUTER SOCIETY

1884 · 1984 THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

Published by IEEE Computer Society Press
1109 Spring Street
Suite 300
Silver Spring, MD 20910

ISBN 0-8186-0512-X (paper)
ISBN 0-8186-4512-1 (microfiche)
ISBN 0-8186-8512-3 (casebound)
Library of Congress Number 83-82834
IEEE Catalog Number 83CH1981-0
IEEE Computer Society Order Number 512

Order from: IEEE Computer Society
Post Office Box 80452
Worldway Postal Center
Los Angeles, CA 90080

IEEE Service Center
445 Hoes Lane
Piscataway, NJ 08854

1884 · 1984 The Institute of Electrical and Electronics Engineers, Inc.

Table of Contents

Session IA
Network Protocols

Chair

P.D. Amer
University of Delaware

SIMULATION OF FARA/CS, A NEW ACCESS PROTOCOL

A.K. Elhakeem,

Elec. Eng. Dept.
Concordia University
Montreal, P.Q. Canada
H3G 1M8

R.K. Goel, and A.P. Dhawan

Elec. Eng. Dept.
University of Manitoba
Winnipeg, Manitoba
R3T 2N2

Abstract

A new computer network access protocol that adapts to the traffic conditions, has been proposed.

This broadcasting protocol is convenient for radio, cable and local area networks. Its delay and throughput characteristics will be proved to be superior compared to existing protocols, especially for a dynamic mix of file-interactive users. By FARA/CS (Frame Adaptable Reservation ALOHA with Carrier Sense), we mean that each node senses the frame time slots, estimates the traffic and sets the transmission strategy accordingly.

Though carrier sensing capability is assumed, the transmission policy at each node is decided upon each frame, thus eliminating much of the propogation problems associated with CS or ETHERNET. Also, to decrease the number of collisions in a frame, each user employs a certain probabilistic rule to randomize the number and locations of his transmissions. Though an involved decision tree is searched by each user, for detecting his transmission slots of each frame, a minimum amount of computations is needed per frame (during the interframe period) making the implementations of the nodes feasible with no waste of the precious capacity.

Through simulation, the maximum useful throughput of the new network came to be very close to .996. The network adapts quickly to the changing traffic conditions and the delay performance is comparable to other networks at low traffic and superior at high traffic.

I. INTRODUCTION

A new hybrid approach to the problem of sharing a broadcasting channel among M user is presented in this paper. The new approach is suitable for a dynamic mix of file-interactive users and for communications mediums with longer propa- gation delay (Satellite channels).

The well known techniques of ALOHA and slotted ALOHA [1] have provided networks capacities of 18.4% and 36.8% respectively, never mentioning the instability and huge delays at higher rates.

Other carrier sense techniques (CSMA) [2] may yield higher capacities (above .8) provided that the maximum channel propagation time is much smaller than the packet time. Also, these techniques suffer [3] the hidden terminal problem and their delay performance degrades at higher input rates. A popular subclass of CSMA is ETHERNET [4] which is very suitable for local area networks with small propagation delays.

For Radio channels reservation ALOHA schemes came also into the picture [5], [6].These techniques look like ALOHA at low rates and like TDMA at higher rates. However, these techniques do not perform very well at low traffic. Other controlled schemes such as BRAM [7], MLMA [8], URN [9] have been proposed and they are amenable to local area networks. The new approach is a hybrid of CSMA, reservation ALOHA, and the URN technique. It is convenient for Satellite channels and/or local area networks. The protocol is designed in such a way to assure a fair share of the channel for all users especially for a dynamic mix of file-interactive users, and it may become one of the approaches to solve the hidden terminal problem.

II. NETWORK DESCRIPTION

We assume a network with M users, time is assumed divided into Frame. Each frame containing M slots. The width of a time slot is equal to the packet length in seconds.

Each user (file or interactive) will decide upon his transmission strategy in the coming frame during a certain small period of time (thinking time) preceeding that frame. To decide upon his action, the user has to listen to the whole previous frame. An energy measurement device will determine the number of collisions in the frame, number of idle slots and, number of reserved slots. Also, from these measurements the location of idle, collided, reserved slots should be known. Each user will estimate the traffic from these measurements, following the decision tree. Since the frame is common to all users, they should arrive at the same estimated traffic. The network can be in either one of three states depending on the number of idle and collided slots in each frame and depending on that state, the user follows the decision tree of Fig. (1).

State 1: is defined by: Number of idle and collided slots $>$ TH_2 where TH_2 is a prescribed threshold = M/3.

State 2: is defined by: TH_2 > Number of idle and collided slots > TH_1 where TH_1 TH_2 (e.g. $TH_1 = M/8$).

State 3: is defined by: Number of idle and collided slotes $< TH_1$.

III. NETWORK PROTOCOL

In State 1 (see Fig. 1) all the users (old or new; old meaning those who had one or more reserved slots) will compete on fair basis for the idle and collided (I+C) slots. The location of the specific slots is selected from the total available (I+C) randomly, while the actual number tried is based on the estimated traffic as will follow later.

Actually what we have in state 1 is a modified Crawther reservation ALOHA [5]. Slots of each frame are denoted with their status, i.e. idle (I) if it was not used, collided (C) if one or two users have transmitted at the same time. Self (S) if the user at hand had successfully transmitted a packet in that slot, others (O) if other users have acquired the slot. As in Crawther ALOHA, each user can reserve more than one slot in each frame, he will keep these slots in the next frame as long as he has same packets to transmit, and other users will respect his right. Moreover, this user is free to capture more slots of the (I+C) pool. However, if the number of reserved slots exceeds his needs at one time, he will lose the difference at the next frame. A user who becomes active has to compete first for the (I+C) slots; once he completes successful transmission, the applicable slots will be recognized and respected in the next frame as (R) slot. For a single file user, it is easy to see that this user can occupy the whole frame and finish transmissions in a very short period of time. For a mix of file and interactive users, equality and fairness will be the issue, i.e. each user will get an equal chance to access the channel irrespective of his input rate. Fig. 2 shows a typical application of state 1 strategy.

In State 2: This state is also a kind of modified Crawther protocol [5]. In this state, new active users, together with the old users that had exactly one reserved slot in the previous frame will compete on fair basis for the (idle and collided) slots. Needless to mentione that the old user will try to gain more slots only if he has more than one packet waiting in his buffer. Also those old users who had more than one reservation inthe previous frame should be too content to try to gain more slots, (even if they had more packets than reservations to transmit). These old users who succeeded in transmitting almost all the packets in their buffer and are left with more reserved slots than they really need, will loose the difference between the number of reservations they had and the number of packets in their buffer. Which slots should he leave is determined by a randomization strategy (A small subroutine such as GGPER routine of the well known IMSL library could be used).

The above strategy keeps a balance of traffic between new and old users in such a way that the new users will not suffer long delays because of the greedy old users.

It is to be noted here that the definition of new and old users is unique to the FARA protocol. Other ALOHA, CS, URN, BRAP, MLMA, ETHERNET,...etc. protocols ignore such distinction, thus allowing one user to capture the channel sometimes on the expense of other new users waiting for the channel. Our technique guarantees the young new users a piece of the action. Fig. 3 shows a typical application of state 2 strategy. To be noted that as the number of active users increases the number of idle slots in each frame decreases and finally the network moves to state 3.

In State-3: (See Fig. 4). In this state the users perform according to a modified Binder algorithm [6]. A new active user in this state will transmit only a certain slot allocated to him. This new user will not compete for the idle and collided slots, but may cause collision in his slot and the collidee will move to his slot in turn. An old user having only one slot (not his allocated slot) will keep it as long as there is no collision in this slot but moves to his allocated slot if there is any collision in his current slot. An old user occupying more than one slot will lose a certain portion (e.g. one third) of those slots for each one (or more collisions) he hears in the frame. However, to help file users to coexist with interactive users in state 3 these same old users can keep one or more reservations in a frame if there is no collisions at all, (In state-3 this will seldomly occur). In any case, old user will not try to gain more slots than they had in the previous frame. He may be content with his reservations, or lose some of them or move to his preassigned slot but never gain more reservations in state 3.

However as more collisions occur in state-3, more old users loose some of their reservations, the number of (I+C) slots increases thus enabling the network to move to state 2 or state 1.

In these states, contention start again for the released (I+C) slots, enabling more users to enter the channel. Eventually as the traffic goes higher, the old greedy users will have almost no slots to lose to the network and the system will be in state 3 most of the time, and if the traffic is high enough, it moves to fixed assignment TDMA as it should.

The transmission strategy is well predicted in state 3. Yes the network may oscillate slightly between states 3,2,1 before going finally to TDMA, but these moves are well fixed and instability problems will never occur with our transmission strategy.

IV. ESTIMATION OF THE TRAFFIC

If we assume that the combined (network) input and retransmissiont raffic is modelled by a Poission distribution, i.e.

$$P(n=K)= \frac{G_0{}^K e^{-G_0}}{K!}; \; K = 0,1,2, \qquad (1)$$

n is the total number of packets per slot. From (1) we can find G_0 as

$$G_0 = -\log_e P(n=0) \qquad (2)$$

New P(n=0) is the probability of generating no packets at all, i.e. the network is idle. This can be estimated as

[P(n=0)]=(number of idle slots of a frame/total number of slots in the frame). The potential traffic to occupy the empty plus collided slots in the next frame is due mainly to the packets that collided in the current frame. To have an estimate of the retransmission traffic, we note that the probability of collision is,
P(B)=1-P(no packets generated)-P(exactly one packet generated)

$$P(B)=1-e^{-G_0}-G_0e^{-G_0} \qquad (3)$$

Denoting the conditional probability of having exactly r packets/collision as (P(A/B)), and following the well known conditional probability rules, we get,

$$P(A/B)=\frac{P(A,B)}{P(B)} \qquad (4)$$

P(exactly r transmissions are involved in each

$$\text{collision}) = \frac{G_0^r e^{-G_0}/r!}{(1-e^{-G_0}-G_0\, e^{-G_0})} \qquad (5)$$

and it follows that the mean number of stations involved in one collision is given by

$$E = G_0 * (1-e^{-G_0})/(1-e^{-G_0}-G_0*e^{-G_0}) \quad (6)$$

If we multiply this by the number of collisions in the current frame, we get (on average) the total number of packets involved in collisions in that frame, i.e. (frame collision traffic), as
G_C = E *(number of collisions in each frame) (7)

Since all the users are measuring the same number of collisions, reservations,...etc., of each frame and the frame is common to all users, G_C will be common to all of them.

Now out of the total number of (I+C) slots of the previous frame, each active user will transmit in only a part equal to

$$[(I+C)* \frac{1}{G_c}] \qquad (8)$$

where [*] stands for the integer part of x.

This strategy guarantees each user, a fair share of the contention slots (I+C).

An active user who has more packets to transmit than reservations will try to acquire more slots according to equation (8). Which slots would he select out of the total possible (I+C) slots will be determined according to the standard randomization subroutine previously outlined.

Equation (8) seems logical by the fact that an increase in G_C and/or decrease in (I+C) should decrease the number of users trying to acquire the (I+C) slots as should be the case. On the other hand, having a surplus in (I+C) and/or a small collision traffic G_C will be a signal to all users to try to acquire more slots if they need.

V. SIMULATION PROCEDURE

The simulation procedure closely followed the decision tree of Fig. 1. However the following have been assumed in the course of simulation:

1- The total number of slots in each frame equals the total number of users (idle & active).
2- Slot time is equal to a single packet transmission time.
3- All the packet arrivals take place during the interframe period. In other words the buffering delay due to packets arriving during the frame period is not considered.
4- The channel throughput (S) and packet retransmission delay (D) performance of the network is averaged over 50 frames.
5- The total input and retransmission traffic is Poission distributed with parameter G.

Before running the simulation program, certain values should be assigned to M, TH1, TH2, the number of slots lost upon collision by an old user in state 3 (Fig. 1), the total traffic F, and the ratio of file to interactive users in the mixed mode case.

For simulating our network, different vectors representing the buffer of each user, the network frame, the transmission policy of each user,.... etc., are initialized and then modified each frame by the new arrivals, successful transmissions, collisions and new reservations,...etc. The standard library Poission subroutine is called once at the beginning of each frame, thus giving a vector and each component of that vector gives the number of packets generated in that specific frame for a specific user. Each user adds his new generated arrivals to whatever he has in the buffer to get the total buffer contents.

The packets are arranged in the buffer to keep the order of arrivals and the receiver transmits top of the line FCFS (first come first served). Each user will have all the information about the previous frame (locations of (C,I,R) stored in a certain vector representing the network status. From this vector, the user can compute everything in equations (1)-(8). This should enable him (also using the standard randomization subroutine) to determine to himself his transmission strategy in the next frame, (is he going to acquire new slots?? which one?? is he going to lose slots??...etc.). Of course to reach at all those decisions, the user has to go through the decision tree of Fig. (1).

Once decisions are made, each user will adjust his transmission vector. The program then combines logically all vectors of all users to arrive at the vector representation network status. At the beginning of the next iteration (frame), and before generating the new arrivals, each user compares what he did (the vector of his transmission decisions in the last frame) to what really happened in the network (network status) and accordingly adjusts his buffer contents, e.g. decrease them by 3 if he really succeeded in getting 3 reservations in the last frame.

To evaluate S,D of the network at a given value of M, each user computes the number of successful transmissions (reservations) he had in the last frame (comparing his transmission strategy vector with the network status vector). Dividing this by M, he gets his S over one frame. Each user will average all S's of all frames and finally the program averages over all users. The same kind of averaging first over all frames and then over all users will also apply to the calculation of the average packet retransmission delay D. However, to compute the delay of a certain packet waiting in the buffer, a certain variable is used to measure the waiting time (in slots) of the head of the line packet (from the time it became h.o.l to the time it is transmitted). To measure this time, a certain counter is incremented by the elapse of each slot of a frame, in the meantime the user transmission vector is compared to the network status vector, if the user hits a successful transmission counting stops and the contents of the counter give the delay of the specific packet. The counter is restarted again to give the delay of the next packet waiting in the buffer, and it is to be noted here that counting of the slots may extend to the next frame or even many frames as long as the H.O.L. packet has not been transmitted. Once all the above has been done, another value of G is picked and the whole procedure repeats.

VI. SIMULATION RESULTS

The values assigned to M (number of users (slots)) were set to: M = 40, M = 60, M = 80. The values of TH1 and TH2 were set at: TH1 = M/10, TH2 = M/3 for some runs and TH1 = M/8, TH2 = M/(2.5) for another set of program runs. In state-3 (see Fig. 1), the old user will lose (1/3) of his reservations upon collision. For the mixed traffic case we assign to α, a value of .9 and .5 where α is the ratio of interactive users to total number of users. For mixed traffic, the useful throughput and delay have been calculated separately for file users and interactive users and then averaged according to: $S=(S_1+S_2)/2$ and $D=(D_1+D_2)/2$. Figs. (5-13) show some of the obtained results while Table 1 shows the transition of the network status for different number of users and different thresholds.

From Table 1, it is clear that for any user population M, and with all the values of thresholds assumed, the network starts at State 1 for low traffic then goes to state 2 as the traffic increases and finally goes to state 3 for high traffic. Fig. 5,6 shows the capacity and delay performance in the single mode case, (no file users). From Fig. 5,6 we see that higher thresholds TH1, TH2 have little effect on the delay or capacity. However, from Figs. (7,8), it is clear that to improve the S,D performance it may be necessary to increase the thresholds to cope with the increase in the number of users (M=80 in Figs. 7,8 while M=60 in Figs. 1,2). Now if the thresholds TH1, TH2 are held fixed and the number of users M varied (according to Figs. 9,10), it is easily seen that the capacity will deteriorate with M increasing especially at high traffic. (Recall that with increasing M, we should increase TH1, TH2).

The delay (Fig.10) also increases with the number of users M (similar to other Reservation Aloha systems). However, the delay almost flattens at S=1, meaning that our network moves to TDMA, and the maximum packet delay will never exceed the frame size. At this value of S, other networks may exhibit excessive delays.

Turning our attention now to the mixed mode case (file & interactive), we see in Fig. 11, the usual inverse relation between througputs of all the interactive users and all file users (S_1M_1) and (S_2M_2) respectively. Interesting to see though that at low values for the total file traffic G_2 the interactive traffic does not have much effect on S_2M_2 especially if M_1S_1 becomes larger.

In Fig. 12 the total number of file users increases to become 1/2 of the user population. Here the file users will get most of the capacity they need (flat line at G_2=.2) irrespective of the traffic of the small traffic interactive users. However, as G_2 (total traffic of file users) builds up then total useful throughput S_2M_2 decreases then rises again (due to moving to state 3, i.e. TDMA). In any case S_2M_2 will be then much less than G_2 (for high G_2) due to collisions with themselves and the interactive users. Fig. 13 shows the useful throughputs S_1M_1 and S_2M_2 and G_2 at high values for the file traffic (G_2=1,2).

It is seen that for α=.5 (user population is divided into two halfes), S_1M_1 and S_2M_2 approaches a value of .5 as G_1 grows meaning equal share of the channel capacity for file and interactive users. With α=.9, the interactive user useful throughput goes to .9, and that of file users gets to .1 meaning again that all users are treated equally whether he is a file or interactive user and whatever were the actual value of traffic offered to the network (G_1 and G_2).

VII. CONCLUSIONS

A new broadcasting protocol local area (and/or Satellite) packet networks has been proposed. Simulation results have proved the excellent throughput and delay performance of the new protocol. It will be interesting to compare the effects of sensing errors (hidden terminal say) on the performance of FARA and ETHERNET (or other CSMA schemes). It is but logical to expect FARA to outperform those schemes by the fact that many observations are taken by each user to decide upon the transmission strategy on the next frame. An error made in the value of (I) and or (C) (because of sensing errors) will not reflect itself in FARA as much as other CSMA techniques. Also the effect of the propagation time should be less in FARA, since the transmission decisions are made each frame thus leaving each user with enough time to sense the channel (even in Satellite networks). The sharing of the capacity inFARA has been proved to be unique in the case of mixed file interactive traffic by deterministically guaranteeing each user a piece of the capacity. The main disadvantage of FARA is the buffer delay, (packets arriving at the middle of the frame has to wait to be served in the next frame). To improve this delay we have to start with a small frame for low traffic and enlarge it for higher input rates (keeping the decision tree the same). This is currently investigated by the authors, together with the possible application to mixed voice data traffic.

REFERENCES

[1] Kleinrock L., Queueing Systems, Vol. 2, Computer Applications, Wiley-Interscience, New York, 1976.

[2] Tobagi, F.A., "Analysis of a two hop centralized Packet Radio Network: Part II Carrier Sense Multiple Access", IEEE Trans. Comm. Vol. COM-28, pp. 208-216, Feb. 1980.

[3] Tobagi, F.A., and L. Kleinrock, "Packet switching in Radio Channels: Part II - The Hidden terminal problem in Carrier Sense Multiple-access and the busy tone solution", IEEE Trans. Comm. COM-23, pp. 1417-1433, (1975).

[4] Metcalfe, R.M., and Boggs, D.R., "ETHERNET: Distributed Packet Switching for local computer networks", Comm, ACM, Vol. 19, pp. 395-404, July 1976.

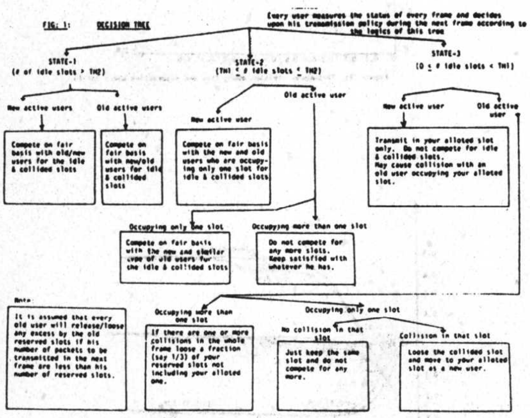
[5] Crawther W., "A system for Broadcast Communication: Reservation Aloha", Proc. of the ninth international conference on system sciences, pp. 371-374, 1973.

[6] Binder, R., "A Dynamic Packet Switchng System for Satellite Broadcast Channels", Proc. ICC, pp. 41-1a to 41-5a, 1975.

[7] Chlamtac, I., Franta, W.R., and Levin, D., "BRAM: The broadcast recognizing access method", IEEE Trans. Comm., Vol. COM-27, pp. 1183-1190, Aug. 1979.

[8] Rothauser, E.H., and Wild, D., "MLMA - A Collision free Multi-Acess method", Proc. IFIP Congr. 77, pp. 431-436, 1977.

[9] Kleinroch, L., and Yemini, Y., "An optimal adaptive scheme for Multiple access Broadcast Communication", Proc. ICC, pp. 7.2.1 to 7.2.5, 1978.



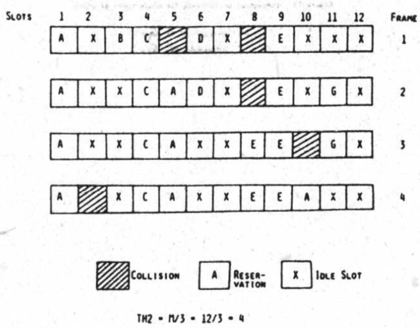


Figure 2: A typical state 1 example

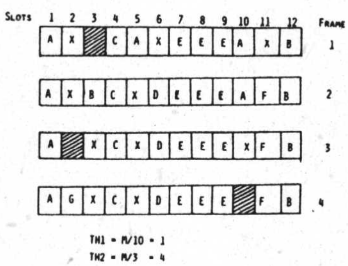


FIGURE 3: A Typical State 2 example

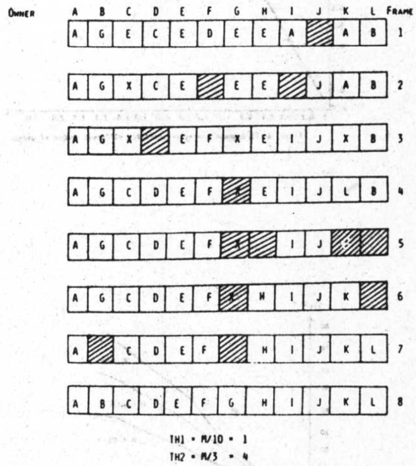


FIGURE 4: A TYPICAL STATE 3 EXAMPLE

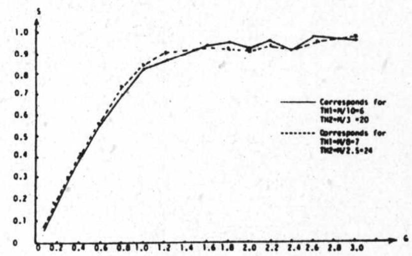


Figure 5: Throughput Characteristics

N = 60 = No. of users = No. of slots in a frame

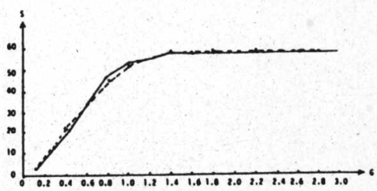


Figure 6: Delay Characteristics

N = 60 = No. of users = No. of slots in a frame

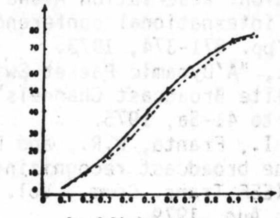


Figure 8: Delay Characteristics
N = 80 = No. of users = No. of slots in a Frame

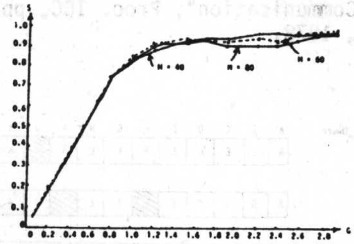


Figure 9: Throughput characteristics for different number of users

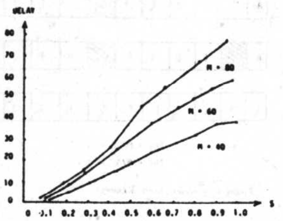


Figure 10: Delay characteristics for different number of users

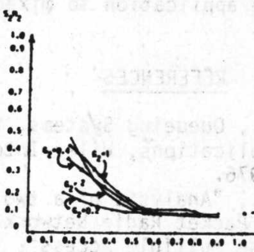


Figure 11: Throughput division among file and interactive users (α=.9)

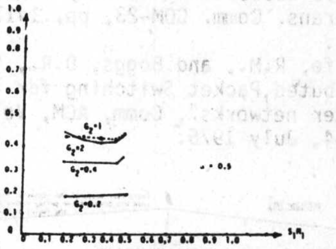


Figure 12: Throughput division among file and interactive users (α=.5)

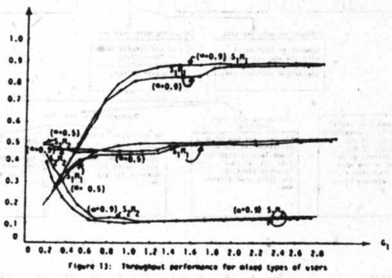


Figure 13: Throughput performance for mixed types of users

corresponds to $G_2 = 1$
corresponds to $G_2 = 2$

USING MARKOV CHAINS TO MODEL COMMUNICATION PROTOCOLS

*Samuel T. Chanson & Arun Kumar**

Dept. of Computer Science
University of British Columbia
Vancouver, B.C., Canada V6T 1W5

ABSTRACT

Mathematical models of communication protocols are generally quite complex. This paper shows that by assuming a packet is equally likely to arrive in any transmission period, then the Markov model, with its simplicity and known solution may be used. We illustrate this by modeling the 1-persistent CSMA protocol and comparing the results with those by Tobagi and Kleinrock [1]. For typical network parameter values, the difference in the results are within 1.6% of one another. The approach may be extended to model other protocols, particularly multiple access network protocols.

INTRODUCTION

The performance characteristics of certain communication protocols, particularly the class of Carrier Sense Multiple Access (CSMA) protocols[2-12], have been analyzed through the use of mathematical models. These models are generally quite complex and their solutions require the use of sophisticated mathematical techniques.

Suppose a packet arrives while some transmission is in progress. It is well known (from the paradox of residual life [4]) that the probability that this packet arrives in a longer transmission period is higher than the probability that it arrives in a shorter transmission period. Analysis of the network taking this fact into consideration accounts for much of the complexity of the model. In this paper, we show that by assuming a packet is equally likely to arrive in any transmission period, then the Markov model, with all its simplicity and known solution, may be used. This assumption is reasonable when the normalized network propagation delay is small. We illustrate this by modeling the 1-persistent CSMA protocol and comparing the results with those by Tobagi and Kleinrock [1]. For a normalized propagation delay (a) of 0.01, which is a typical value for many practical systems, the results obtained using this model are within 1.6% of those obtained using Tobagi's analysis when the offered traffic (G) is less than 5. Even when a is increased by a factor of 10, the difference is still acceptably small. This approach may be extended to model other multiple access network protocols.

* Arun Kumar is now with the Prime Computer Inc., Framingham, Ma 01701, U.S.A.

CH1981-0/83/0000/0009$01.00© 1983 IEEE

1-PERSISTENT CSMA PROTOCOL

This protocol is an improvement over the pure Aloha system first proposed by Abramson [1]. In the Aloha system, the network consists of a number of terminals (or computers) connected by a transmission cable (channel). Whenever a terminal has a packet to transmit, it transmits the whole packet. If while this terminal is transmitting another terminal starts to transmit also, a collision occurs and the information is destroyed. The stations involved wait a random amount of time (depending on the retransmission policy being used) before trying again.

If the terminals are relatively close to one another so that the propagation delay is short compared to the packet transmission time, a terminal can sense the channel for the presence of carrier *before* transmitting a packet. This can significantly reduce the number of collisions and thus improve the channel utilization. Such protocols in which a terminal listens for the carrier before transmitting are known as Carrier Sense Multiple Access protocols (CSMA).

The 1-persistent CSMA is a member of the CSMA protocols in which a ready terminal, after sensing the channel behaves as follows:

- If the channel is sensed to be idle, it transmits the packet immediately with probability one.

  If the channel is sensed to be busy, it continues to sense the channel until it becomes idle and then transmits the packet. That is, it persists in transmitting and is therefore known as 1-persistent CSMA.

THE MODEL

Consider the packet transmission time to be one unit and the end to end propagation delay to be 'a' units (all units of time are normalised by the packet transmission time). To simplify the analysis we assume that the propagation delay between any two stations is 'a' units. This assumption gives a lower bound on the throughput.

Let t denote the time a packet is transmitted immediately upon arrival into an idle channel. If another packet arrives between t and t+a, the channel will still appear to be idle and this packet will also be transmitted. This will create a collision. If no packet arrives during t and t+a, then the first packet will be successfully transmitted.

In the event of a collision, let t+Y be the time of arrival of the last packet arriving between t and t+a (see Fig. 1). Thus, the length of a successful transmission period is 1+a and the length of an unsuccessful transmission period is 1+Y+a.

Any packet arriving after the first a seconds of a transmission period will sense the channel to be busy and must wait until the channel is sensed idle, at which time they all would be transmitted simultaneously. We assume that the arrival rate of both new and rescheduled packets has a poisson distribution and we represent this arrival rate by G.

Let us calculate $\bar{Y}$, the mean value of Y.

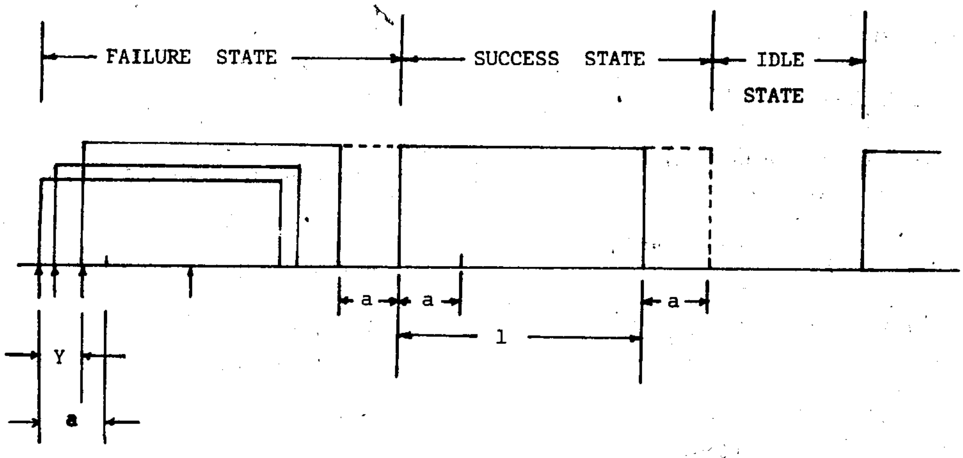


Figure 1. State Timing Diagram

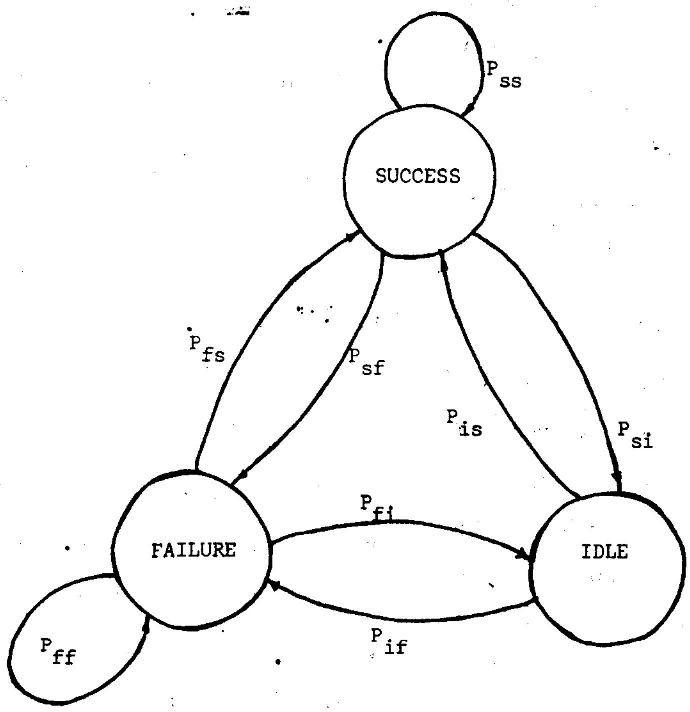


Figure 2. Markov Model

Pr(Y≤y) = Pr (at least one arrival occurs in the first y seconds and no arrival occurs during the next a-y seconds| at least one arrival occurs in the first a seconds).

$$= \frac{\exp(-G*(a-y))*(1-\exp(-G*y))}{1-\exp(-a*G)}$$

$$Pr(Y>y) = 1-\frac{(\exp(-G*(a-y)-\exp(-a*G))}{1-\exp(-a*G)}$$

$$= \frac{1}{1-\exp(-a*G)} * (1-\exp(-G*(a-y)))$$

$$\text{or } \overline{Y} = \int_{y=0}^{y=a} Pr(Y>y)dy = \frac{1}{1-\exp(-a*G)} * (a-(1/G)*(1-\exp(-a*G))) \qquad (1)$$

The detailed analysis of 1-persistent CSMA protocol is complicated by the fact that a packet is more likely to arrive in a longer transmission period than in a shorter one. This implies that the distribution of the lengths of transmission period needs to be calculated. We can, however, greatly simplify the analysis without sacrificing accuracy for networks with small end to end propagation delay if we ignore the distribution of the lengths of unsuccessful transmission periods.

Let us construct a Markov's model with three states (Fig. 2): Success state, Failure state and Idle state. The Success state represents a successful transmission, The Failure state represents an unsuccessful transmission and the Idle state corresponds to an idle channel. The state transition probabilities are quite obvious (that is, their derivation does not require much effort) and are shown below:

P_{is} = Pr(transition from Idle to Success state)
= Pr(no packet arrives during the first a seconds)
= exp(-a*G) (2)

P_{if} = Pr(transition from Idle to Failure state)
= $1 - P_{is}$ (3)

P_{si} = Pr(transition from Success to Idle state)
= Pr(no arrival during packet transmission time)
= exp(-G) (4)

P_{ss} = Pr(transition from Success to Success state)
= Pr(one arrival during the packet transmission time) * Pr(no arrival during the next a seconds)
= G*exp(-G)*exp(-a*G)
= G*exp(-(1+a)*G) (5)

P_{sf} = Pr(transition from Success to Failure state)
= $1 - P_{si} - P_{ss}$ (6)

P_{fi} = Pr(transition from Failure to Idle state)
= Pr(no arrival during an unsuccessful transmission period)
= $\exp(-(\overline{Y}+1)*G)$ (7)

P_{fs} = Pr(transition from Failure to Success state)
= Pr(one packet arrives during the unsuccessful transmission period) * Pr(no packet arrives during the next a second)
= $(\overline{Y}+1)*G*\exp(-(\overline{Y}+1)*G)*\exp(-a*G)$
= $(\overline{Y}+1)*G*\exp(-(\overline{Y}+1+a)*G)$ (8)

P_{ff} = Pr(transition from Failure to Failure state)
= $1 - P_{fi} - P_{fs}$ (9)

Let P_s,P_f and P_i be the probability of being in Success, Failure and Idle state respectively. These probabilities are related to the state transition probabilities by the following set of equations.

$$P_s = P_s*P_{ss} + P_f*P_{fs} + P_i*P_{is} \qquad (10)$$

$$P_f = P_s*P_{sf} + P_f*P_{ff} + P_i*P_{if} \qquad (11)$$

$$P_i = P_s*P_{si} + P_f*P_{fi} \qquad (12)$$

Only two out of the above three equations are independent. Therefore we introduce another constraint: