A COMMUNICATION NETWORK WITH SOME HIGH PRIORITY MESSAGES

I. Stavrakakis, D. Kazakos
Department of Electrical Engineering
University of Virginia
Charlottesville, VA 22901

Abstract

In this paper we consider a communication system with a large number of geographically separated users. An active user (i.e. one who needs to make use of the common communications resource) can be in one of two possible states (H, L), depending on his nature and / or the type of information that he desires to communicate. Users who are in state H are given some priority over those in state L. Users which are in state H by nature can be commanders in a military environment. Users which are in state H due to the type of information that they possess can be any user who has a critical information that deserves fast transmission to a central decision maker. The traffic generated by the users in state H is assumed to be small compared to the total traffic that the system can accommodate.

For the above system we develop a binary-feedback (collision / non-collision) random-access protocol which serves all the users. Throughput analysis is performed and the stability region of the system is obtained. Mean packet delay results are also analytically obtained for the cases in which the traffic generated by the users in state H is less than 30% of the total traffic that the system can accommodate. The delay results show that the protocol induces much shorter delay for the high priority messages.

I. Introduction

A lot of work has been directed towards the development of the multi user random access communication systems with a homogeneous population of users [1]-[5]. There are many practical applications, however, in which some or all users can alternate between to possible states H and L. Packets which are generated by users in state H should be given some priority over those generated by users in state L. Users who are in the same state are considered to be in the same class. As a result two classes of users are created (H, L) and the user population is generally non-homogeneous.

In a military environment, permanent members of class H can be commanders, while any other user of the system who has critical information can move from class L to class H temporarily and return to his original class after the critical information has been transmitted successfully.

In a mobile user environment where users move in and out of the range of the system, or move from region to region, fast moving users may need to experience shorter delays than the regular ones; this may be necessary to make packet transmission possible while the user is still inside the region. Also, users that are close to the boundaries of a region and are going to move outside it, should experience shorter delays. These users can be members of class H.

In a non-military static user environment members of class H can be users who pay more or users who carry control information which is critical for the operation of a system. have high priority and should reach their destination faster than the regular ones. High priority packets can be those which are generated by high priority users (e.g. important users, or users that can pay more for better service), or can be packets that are generated by any user of the system but the information that is carried is characterized as important and deserves high priority in its transmission.

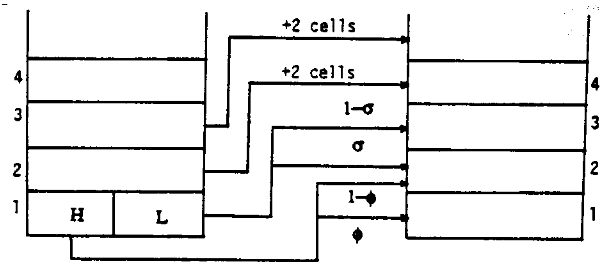
In the next section we describe a communication system with two classes of messages (or users) in detail. In the same section we also describe the proposed random access protocol which determines the common channel alocation. In section III throughput and delay analysis are briefly described, while in the last section the results of the analysis are shown and conclusions are drawn.

II. The Random Access Protocol

We consider a large population of geographically separated users who use a single communication channel. Users which at certain time instant are in state H have some priority over the rest of the population and they form the high priority class H. It is assumed that the packet traffic generated by that class represents only a small percentage of the total traffic that is served by the system. In other words, it is assumed that the packets that need special service are rare and this is a realistic assumption at least for the environments which were described above.

Research supported by the Air Force Office of Scientific Research through Grant AFOSR 87-0095.

(a) If F = C then



(b) If F = NC then

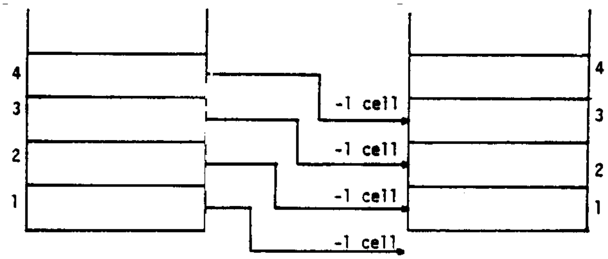


Figure 1.
Operation of the algorithm via the imaginary stack;
H (L) are the high (low) priority users.

session length of multiplicity (mu , nu), by following procedures similar to those that can be found in [4], [7], [8]. The set of pairs (λ_h, λ_l) for which such a bound was possible to obtain, is a lower bound on the stability region of the algorithm. An upper bound can be obtained by solving a truncated version of an infinite dimensionality linear system of equations with respect to $L_{\mu,\nu}$, [14]. The latter system is obtained by considering the expectations of the recursive equations which describe the operation of the system. The stability region of the algorithm is plotted and it is shown in Fig. 2.

The mean delay of the high and low priority packets is also calculated but only for input traffic pairs (λ_f, λ_s) such that $\lambda_f \le .065$ packets per packet length. For that region, bounds on the involved quantities was possible to obtain. This range of pairs determines the operation region of the algorithm; i.e.

$$S_{op} = \left\{ (\lambda_h, \lambda_l) : 0 \le \lambda_h \le 0.065,\ 0 \le \lambda_l \le \lambda_{l,max}(\lambda_l) \right\}$$

where $\lambda_{l,max}(\lambda_h)$ can be obtained from Fig. 2. The delay analysis is performed by applying the regeneration theory procedures that appear in [12], [4], [9], [10], or by using directly the strong law of large numbers, [11], [7]. Very tight upper and lower bounds on the mean delay of the high and low priority packets, D_h and D_l, respectively, were calculated for some values of the input traffic; the results appear in Table 1.

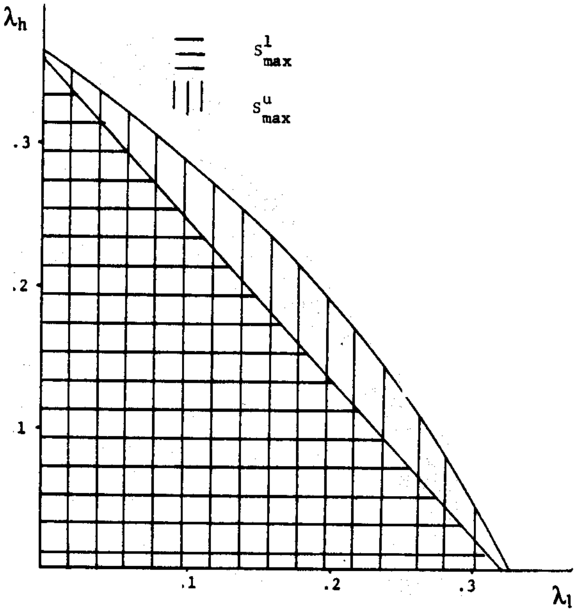


Figure 2.
Upper and lower bounds on the maximum stable throughput; λ_h and λ_l are in packets per packet length.

V. Results and Conclusions

The protocol that we developed and analyzed is appropriate for an environment where users can be in one out of two possible states. Two different classes of users are created to acommodate users in different states. Thus the user population is non-homogeneous and some user are given priority to transmit their packet. An algorithm for a homogeneous user population that consists of users in state H only and use binary feedback information and simple splitting after a collision, has been found to achieve a maximum stable throughput of ~ .36 [13]. The algorithm that we suggest for the non-homogeneous population achieves total throughput, at least, between .320 - .357 depending on the contribution of the two classes to the total input traffic.

In Fig. 3, Fig. 4 and Fig. 5, plots of the bounds on D_h and D_l versus λ_l, for λ_h=0.01, λ_h=0.03 and λ_h=0.065 respectively, are shown. These values of λ_h correspond to an input traffic coming from the high priority class equal to ~ 3%, ~ 10% and ~ 20% of the total traffic that can be served by the system. From the plots it can be observed that the high priority packets (coming from users in state H) experience shorter delays than the packets of class *L* ; the difference is essential for $\lambda_l > .5\lambda_{l,max}$. If the nominal

The input traffic to the channel that is generated by each class of users is assumed to be Poisson distributed with intensities λ_h and λ_l respectively; the Poisson model is proved to be an appropriate model for the cumulative traffic that is generated by a large population of bursty users, which is assumed to be the case in the system under consideration. Messages are assumed to be packetized and of fixed length; it is assumed that time axis is slotted and that the beginning of a packet transmission coincides with the beginning of a slot.

All users may access the channel as long as they have a packet to transmit; the first transmission attempt takes place at the beginning of the first time slot that follows the packet generation instant. Because of the freedom that the users enjoy in accessing the channel, a transmission attempt results in either a successful packet transmission, or in a packet collision if more than one packet transmissions were attempted in the same time slot. Thus it becomes obvious that an algorithm is necessary in order for the conflicts to be resolved and the channel to remain usable.

It is assumed that all users that have a packet to transmit (and only these users need to do that) keep sensing the channel and are capable of detecting a packet collision; that is, we assume that a binary feedback information is available to all active users before the end of the current slot, revealing whether the slot was involved in a packet collision (C) or not (NC). Channel errors are not taken into consideration and packet collision is the only event that results in an unsuccessful transmission.

The first time transmission policy is kept the same for both classes of users; it is simple and implies that a packet is transmitted at the beginning of the first slot following the packet generation instant. It is apparent that if the two classes are to experience different delays, they should follow different steps in the collision resolution procedure. We develop a simple limited sensing collision resolution algorithm. The limited sensing characteristic is apparently important for a mobile user environment since the users may not be able to know the history of the channel before their packet generation instant. We assume that the state of a user is determined by the content of a counter that is assigned to each one of them; this counter is updated according to the steps of the algorithm and the feedback from the channel. Users whose counter content at the beginning of a time slot is equal to one, transmit in that slot.

Let $c_i^f (c_i^s)$ denote the counter content of a high priority (regular) user, at the beginning of the i^th^ time slot. Let also F_i, $F_i \varepsilon (C, NC)$, denote the channel feedback information just before the end of the i^th^ time slot. The steps of the collision resolution algorithm consist of the following counter updating procedures that take place at the end of each time slot.

(A) If $F_i = C$ then

$$c_i^h = 1 \begin{array}{l} \nearrow c_{i+1}^h=1 \text{ with probability } \phi \\ \searrow c_{i+1}^h=2 \text{ with probability } 1-\phi \end{array}$$

$$c_i^l = 1 \begin{array}{l} \nearrow c_{i+1}^l=2 \text{ with probability } \sigma \\ \searrow c_{i+1}^l=3 \text{ with probability } 1-\sigma \end{array}$$

$$c_i^j = r \rightarrow c_{i+1}^j = r+2 , \;\; r \geq 2 , \; j \varepsilon (l,h)$$

(B) If $F_i = NC$ then

$$c_i^j = r \rightarrow c_{i+1}^j = r-1 , \;\; r \geq 1 , \; j \varepsilon (l,h)$$

The first time transmission policy can also be described by using the concept of the counter; it simply implies that a new user sets the counter equal to one at the end of the slot in which its packet arrival took place. It did not seem to us reasonable to develop different first time transmission policies for the two classes of users. It would probably be a waste of the channel capacity to give priority to rarely appearing high priority packets, before it becomes known that a collision took place. If a conflict occurs, then the collision resolution algorithm offers some priority to the high priority packets that were involved in the conflict.

From the description of the algorithm it can be easily observed that the system is of continuous entry, i.e. new users enter the system at the beginning of the first slot that follows their packet arrival, unlike what happens in the blocked access algorithms [2]. The limited sensing characteristic of the algorithm, together with the lack of need for a central controller to coordinate the users, increase the robustness and applicability of the system.

The operation of the algorithm can also be described via the concept of the imaginary stack. Users whose counter content equals n are located in the n^th^ cell of the stack. Depending on the channel feedback and their location in the stack, the users move up and down as it is shown in Fig. 1.

IV. Performance Analysis of the Protocol

In this section we derive bounds on the stability region of the algorithm and the mean packet delay. Analysis is based on the concept of the session and the development of recursive equations to describe the operation of the system. A session is defined as a number of consecutive slots between properly selected renewal points of the system, [6]. If μ users in state H and ν users in state L attempted a packet transmission in the first slot of a session, then the pair (μ, ν) determines the multiplicity of that session. It can be easily concluded that the multiplicities of the sessions are independent identically distributed random variables.

If for an input traffic pair (λ_f, λ_s), the expected value of the session length of multiplicity (μ, ν) is finite, for μ and ν finite, then we say that the operation of the system is stable and the pair (λ_f, λ_s) belongs to the stability region of the system. The maximum overall sets of stable points (λ_h, λ_l) determines the maximum stable throughput region and is denoted by S_{max}.

Detailed stability and delay analysis of the proposed protocol can be found in [6]. For the stability analysis of the system we calculate a linear upper bound on the mean point of operation of the system is set around $\lambda_l = .9\lambda_{l,max}$, then the average high priority packet delay is less than half the one of the other class.

In table 1, the delay results of the suggested algorithm are compared with the delay, D*, that the homogeneous class equivalent algorithm (as described above), induces [13]. Again we can observe that always $D_h < D^*$ and particularly $D_h < .5D^*$ around the nominal point, the latter being defined as before.

Since privileged service is offered to some users, there has to be a price that the rest of the population must pay. The first consequence is the small reduction in the total throughput, as mentioned before. The other penalty is the increased average low priority packet delay compared with the one that the homogeneous population equivalent algorithm induces. From table 1 we can see that, indeed, $D_l > D^*$, as it was expected. The increase in D_l is far from catastrophic and it is realistic to consider that it is possible for a system to tolerate these delay increases for the low priority class, especially if strict limitations exist for the high priority users.

λ_h	λ_T	λ_l	D_h	D_l	D^*
.01	.02	.01	1.555	1.590	~ 1.57
	.11	.10	1.829	2.369	~ 2.10
	.18	.17	2.186	3.815	~ 2.90
	.26	.25	3.095	9.922	~ 6.20
	.31	.30	5.793	39.793	~ 16.00
	.32	.31	8.718	78.748	~ 23.00
.03	.04	.01	1.632	1.678	~ 1.66
	.13	.10	1.951	2.571	~ 2.21
	.20	.17	2.389	4.312	~ 3.33
	.28	.25	3.672	12.681	~ 8.33
	.31	.28	5.453	28.961	~ 16.00
	.32	.29	7.113	45.905	~ 23.00
.065	.075	.01	1.800	1.878	~ 1.82
	.165	.10	2.234	3.054	~ 2.70
	.235	.17	2.900	5.595	~ 4.33
	.315	.25	5.801	23.101	~ 18.00
	.325	.26	7.200	33.080	~ 26.00

Table 1.
Delay results.

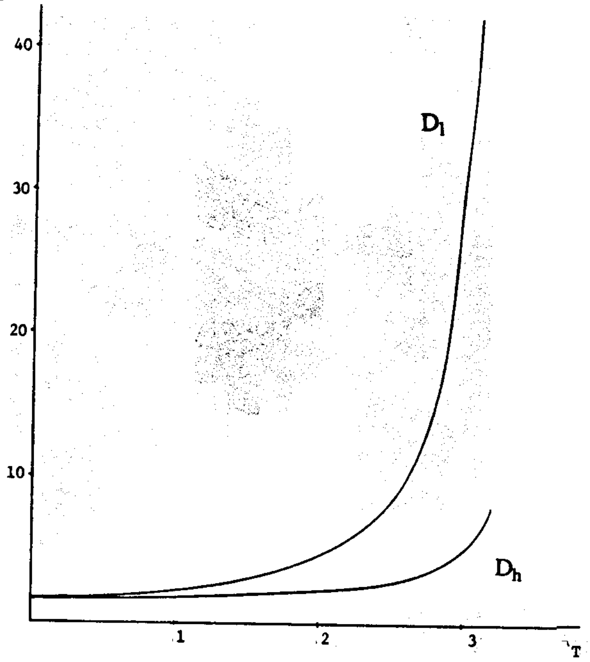


Figure 3.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for $\lambda_h = .01$ (packets/packet length).

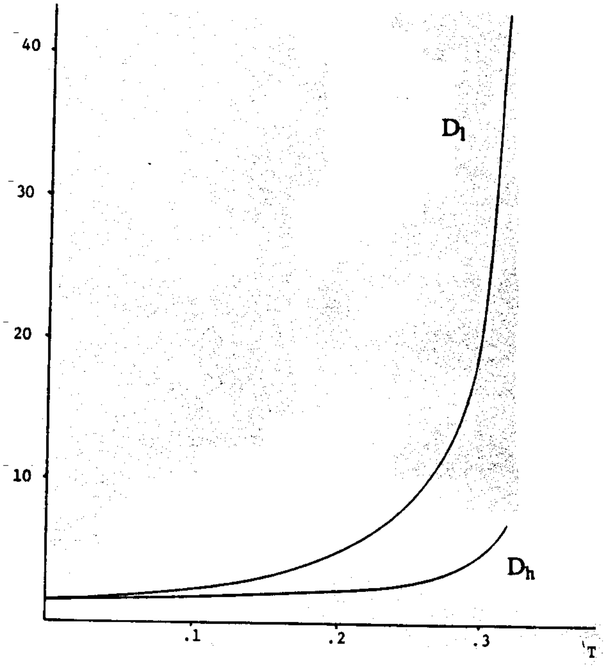


Figure 4.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for $\lambda_h = .03$ (packets/packet length).

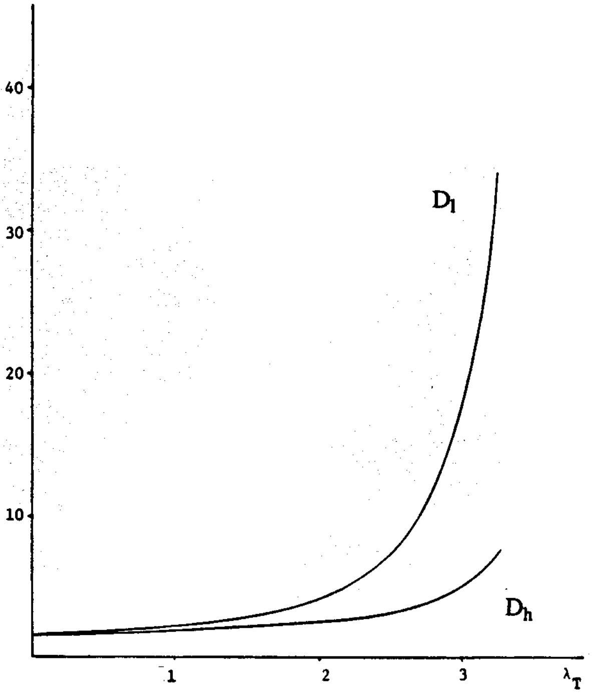


Figure 5.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for λ_h=.065 (packets/packet length).

References

[1] B.S. Tsybakov, N.D. Vvedenskaya, "Random Multiple Access Stack Algorithm", translated from Problemy Peredachi Informatsii, Vol. 16, No. 3, pp. 80-94, July-September 1980.

[2] J.I. Capetanakis, "Tree Algorithm for Packet Broadcast Channel", IEEE Transactions on Information Theory, Vol. IT-25, No. 5, pp. 505-515, Sept. 1979.

[3] R.G. Gallager, "Conflict Resolution in Random Access Broadcast Networks", Proc. AFOSR Workshop on Commun. Theory and Applications, Provincetown, MA, pp. 74-76, Sept. 1978.

[4] L. Georgiadis, P. Papantoni-Kazakos, "Limited Feedback Sensing Algorithms for the Packet Broadcast Channel", IEEE Transactions on Information Theory, Special Issue on Random Access Communications, Vol. IT-31, No. 2, pp. 280-294, March 1985.

[5] N. Abramson, "The ALOHA System - Another Alternative for Computer Communications", Proc. AFIPS Fall Joint Computer Conference, Houston, Texas, Nov. 17-19, 1970, AFIPS Press, Montvale, N.J., pp. 281-285.

[6] J. Kurose, M. Schwartz, Y. Yemini, "Multiple Access Protocols and Time Constrained Communication", ACM Computing Surveys, Vol. 16, No. 1, March 1984.

[7] I. Stavrakakis, D. Kazakos, "A Simple Stack Algorithm for a Code Division Multiple Access Communication System", Technical Report, No. UVA/525656/EE87/101, University of Virginia, October 1986.

[8] N.D. Vvedeuskaya, B.S. Tsybakov, "Random Multiple Access of Packets to a Channel with Errors", Problemi Peredachi Informatsii, Vol. 19, No. 2, pp. 52-68, April-June 1983.

[9] J.W. Cohen, "On Regenerative Processes in Queueing Theory", New York: Springer-Verlag, 1976.

[10] S. Stidham, Jr., "Regenerative Processes in the Theory of Queues, with Applications to the Alternating-Priority Queue", Adv. Appl. Prob., Vol. 4, pp. 542-577, 1972.

[11] G.L. Chung, "A course in probability theory", Academic Press, Inc. 1974.

[12] L. Georgiadis, L. Merakos, P. Papantoni-Kazakos, "Unified Method for Delay Analysis of Random Multiple Access Algorithms", Technical Report, UCT/DEECS/TR-85-8, University of Connecticut, Aug. 1985. Also submitted for publication.

[13] P. Mathys, "Analysis of Random Access Algorithms", Ph.D. dissertation, Diss. ETH No. 7713, Swiss Federal Institute of Technology (ETH), Zurich, 1984.

[14] L.V. Kantorovich, V.I. Krylov, "Approximate methods of higher analysis", pp. 21, Interscience Publishers, 1958.