A Multiuser Random-Access Communication System for Users with Different Priorities

Ioannis Stavrakakis and Demetrios Kazakos

***Abstract*—In this paper a binary feedback collision resolution algorithm is developed for a multiuser random access communication system with nonhomogeneous user population. The user population is split into two classes with different priorities. Throughput and delay analysis of the proposed algorithm are performed and numerical results are obtained.**

I. Introduction
Description of the System and the Algorithm

Most of the existing literature on the multiuser random access communication systems assumes homogeneous user population [1]. There are many practical applications, however, where it is desired that packets generated by certain users experience shorter delays than the regular packet of the system [2]. Under these conditions, different priority classes of users should be considered. In this paper, the user population supported by a single communication channel is split into two priority classes; the high priority class, which generates packets according to a Poisson process with rate λ_f and the low priority class, which generates packets according to a Poisson process with rate λ_s. Information is packetized and has fixed length. The time axis is slotted; a slot is equal to the packet transmission time. A packet transmission may be attempted only at the beginning of a slot.

An active user (that is, a user with a packet ready for transmission) attempts the first packet transmission at the beginning of the first time slot that follows its packet generation instant. This first time transmission policy, which is the same for both classes, is reasonable since it might be a waste of the channel capacity to give priority to high priority packets (and delay the low priority ones) before it becomes known that more than one packets currently compete for the channel. A transmission attempt may result in either a successful transmission or in a packet collision, if more than one packet transmissions are attempted in the same time slot. To resolve the packet conflicts, a collision resolution algorithm is necessary. This algorithm also implements the priority scheme that results in lower packet delay for the high priority users. It is assumed that all active users (and only these users need to do so) are capable of sensing the channel and detecting a packet collision. That is, 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). The channel is assumed to be error-free.

A simple stack algorithm [1], is developed for the resolution of the packet conflicts. The limited sensing characteristic is important when users do not have a record of the past channel history before the packet generation instant. The limited sensing characteristic of the algorithm, together with the distributed nature of the protocol, increase the robustness and the applicability of the system. The state of a user is assumed to be determined by the content of a counter; this counter is updated according to the steps of the algorithm and the feedback from the channel. Users whose counter content is equal to one at the beginning of a time slot transmit in that slot. A currently activated user sets its counter equal to one and attempts transmission. Let c_i^f (c_i^s) denote the counter content of a high priority (regular) user, at the beginning of the ith time slot. Let F_i, $F_i \in (C, NC)$, denote the channel feedback information just before the end of the ith 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) Let $F_i = C$: if $c_i^f = 1$ then $c_{i+1}^f = 1$ with probability ϕ and $c_{i+1}^f = 2$ with probability $1-\phi$; if $c_i^s = 1$ then $c_{i+1}^s = 2$ with probability σ and $c_{i+1}^s = 3$ with probability $1-\sigma$; if $c_i^j = r$ then $c_{i+1}^j = r+2$, $r \geq 2$, $j \in (s, f)$. (b) Let $F_i = NC$: if $c_i^j = r$ then $c_{i+1}^j = r-1$, $r \geq 1$, $j \in (s, f)$.

II. Throughput/Delay Analysis and Numerical Results

The details of the analysis can be found in [2]. The concept of the session is useful for the development of recursive equations which describe the operation of the system. A session is a sequence of consecutive slots defined as follows: A session of multiplicity (μ, ν), $\mu \geq 0$, $\nu \geq 0$ and $\mu + \nu > 1$, is defined by using the concept of a virtual stack and a marker; the stack is assumed to have infinite number of cells; the marker is placed in cell 0 at the time origin. The first slot involved in a packet collision marks the beginning of a session of multiplicity (μ, ν), if μ high priority and ν low priority packets were involved in that original collision. At this time the marker is placed in either cell 3 or cell 2, depending on whether a low priority user was involved in that conflict, or not. In the sequel, the marker moves two cells upwards or one cell downwards, depending on whether the feedback was C or NC, respectively. The movement of the marker takes place at

Paper approved by the Editor for Network Protocols of the IEEE Communications Society. Manuscript received January 10, 1989; revised October 2, 1990. This work was supported by the Air Force Office of Scientific Research under Grant AFOSR 82-0030 and the U.S. Office of Naval Research under Contract ONR-N14-86-K-0742. This paper was presented in part at the 26th Conference on Decision and Control (CDC), Los Angeles, CA, December 9–11, 1987.

I. Stavrakakis is with the Department of Electrical Engineering and Computer Science, University of Vermont, Burlington, VT 05405.

D. Kazakos is with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22901.

IEEE Log Number 9102282.

the end of a slot. The slot in which the marker moves to cell 0, is the last slot of the session. The first slot involved in a collision that follows, marks the beginning of another session of multiplicity (μ, ν), $\mu \geq 0$, $\nu \geq 0$ and $\mu + \nu > 1$, if μ high priority and ν low priority packets are collided. Sessions of multiplicity (μ, ν), $\mu + \nu \leq 1$, result in no movement of the marker and are defined to have length equal to one time slot. It should be noted that sessions cannot be identified by the users and that they are used only in the analysis of the operation of the system. The multiplicities of the sessions are independent identically distributed random variables with joint probability determined by the product of two Poisson probabilities with rates λ_f and λ_s. The stability region, $S_{\max}$, of the system is defined as the maximum over all sets of rates (λ_f, λ_s) which result in finite mean session length of multiplicity (μ, ν), $\tau(\mu, \nu)$, for μ and ν finite. The following equations can be derived:

$$\begin{aligned} &\tau_{0,0} = \tau_{0,1} = \tau_{1,0} = 1, \\ &\tau_{\mu,\nu} = 1 + \tau_{\phi_1+F_1,S_1} + \tau_{\mu-\phi_1+F_2,\sigma_1+S_2} + \tau_{F_3,\nu-\sigma_1+S_3}, \\ &\qquad \mu \geq 1,\ \nu \geq 1 \end{aligned} \tag{1a}$$

$$\begin{aligned} &\tau_{\mu,0} = 1 + \tau_{\phi_1+F_1,S_1} + \tau_{\mu-\phi_1+F_2,S_2}, \qquad \mu \geq 2, \\ &\tau_{0,\nu} = 1 + \tau_{F_1,S_1} + \tau_{F_2,\sigma_1+S_2} + \tau_{F_3,\nu-\sigma_1+S_3}, \\ &\qquad \nu \geq 2, \end{aligned} \tag{1b}$$

where $F_i, S_i, i = 1,2,3$, are independent Poisson distributed random variables with parameters λ_f and λ_s respectively; ϕ_1, σ_1 are independent random variables that follow the binomial distribution with parameters μ, ϕ and ν, σ respectively. Let $L_{\mu,\nu}$ denote the expected value of $\tau(\mu, \nu)$. By considering expectations in (1), an infinite dimensional linear system of equations, of the form

$$L_{\mu,\nu} = h_{\mu,\nu} + \sum_{k=0}^{\infty}\sum_{j=0}^{\infty} a_{k,j}^{\mu,\nu} L_{k,j}, \qquad \mu \geq 0, \qquad \nu \geq 0. \tag{2}$$

is obtained with respect to $L_{\mu,\nu}$; the constants can be found in [2]. The stability region of the algorithm can be established by deriving bounds on the solutions of (2). The approach taken is presented in [2] and it is similar to that appearing in [3]. The derived bounds on $S_{\max}$ are plotted in Fig. 1. Notice that when only the high priority class is present, the system seems to be capable of supporting larger traffic ($\lambda_f \leq 0.357$) than in the case when only low priority users are present ($\lambda_s \leq 0.32$). This is not surprising since although the algorithmic structure is the same for any single class, there is always a waste of the first slot after a collision among low priority users. As a consequence, a reduction in the maximum stable throughput, when only low priority users are present, is expected.

Under stability conditions, the existence of renewal slots, which mark the beginning of statistically identical sessions, implies that the operation of the system can be described by a regenerative process. As a result, conclusions about the

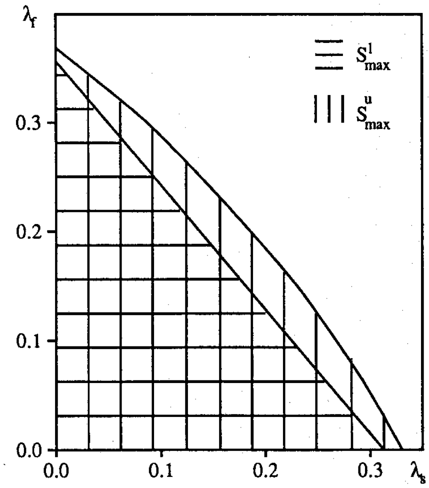

Fig. 1. Upper, $S^u_{\max}$, and lower, $S^l_{\max}$, bounds on the maximum stable throughput.

limiting behavior of the system can be drawn through the manipulation of quantities that are defined on a session. Let W_f and W_s be the mean cumulative delay of all high and low priority packets, respectively, which are transmitted in a single session; the packet delay until the beginning of the time slot that follows the packet arrival is not included in W_f or W_s. If D_f and D_s denote the average delay of a high priority and a regular packet respectively, then the following bounds can be established, in view of the above discussion:

$$\begin{aligned} 0.5 + \frac{W_f^l}{\lambda_f L^u} \leq D_f \leq 0.5 + \frac{W_f^u}{\lambda_f L^l}, \\ 0.5 + \frac{W_s^l}{\lambda_s L^u} \leq D_s \leq 0.5 + \frac{W_s^u}{\lambda_s L^l} \end{aligned} \tag{3}$$

where W_f^l, W_s^l, L^l and W_f^u, W_s^u, L^u denote lower and upper bounds on the corresponding quantities; L is the average session length and 0.5 is the mean packet delay until the beginning of the first time slot that follows the packet arrival. The bounds on L are derived as shown in [2], in terms of the bounds on $L_{\mu,\nu}$. To derive bounds on W_f and W_s a similar approach is followed where $\tau_{\mu,\nu}$ is replaced by $\omega_{\mu,\nu}^f$ and $\omega_{\mu,\nu}^s$ which are the cumulative delays of the high and the low priority packets, respectively, which are transmitted during a session of multiplicity (μ, ν). Similarly to (1), $\omega_{\mu,\nu}^f$ and $\omega_{\mu,\nu}^s$ can be shown to satisfy the following recursive equations:

$$\begin{aligned} &\omega_{0,0}^f = \omega_{0,1}^f = 0, \qquad \omega_{1,0}^f = 1 \\ &\omega_{\mu,\nu}^f = \mu + \omega_{\phi_1+F_1,S_1}^f + (\mu - \phi_1)\tau_{\phi_1+F_1,S_1} + \omega_{\mu-\phi_1+F_2,\sigma_1+S_2}^f \\ &\qquad + \omega_{F_3,\nu-\sigma_1+S_3}^f \ \mu \geq 1, \quad \nu \geq 1. \end{aligned} \tag{4a}$$

$$\begin{aligned} &\omega_{\mu,0}^f = \mu + \omega_{F_1+\phi_1,S_1}^f + (\mu - \phi_1)\tau_{F_1+\phi_1,S_1} + \omega_{\mu-\phi_1+F_2,S_2}, \\ &\qquad \mu \geq 2; \\ &\omega_{0,\nu}^f = \omega_{F_1,S_1}^f + \omega_{F_2,\sigma_1+S_2}^f + \omega_{F_3,\nu-\sigma_1+S_3}, \quad \nu \geq 2. \end{aligned} \tag{4b}$$

TABLE I

λ_f	λ_T	λ_s	$L^l \sim L^u$	$W_f^l \sim W_f^u$	$W_s^l \sim W_s^u$	$D_f^l \sim D_f^u$	$D_s^l \sim D_s^u$	D^*
0.01	0.02	0.01	1.000	0.010	0.010	1.555	1.590	~1.57
	0.11	0.10	1.046	0.013	0.195	1.829	2.369	~2.10
	0.18	0.17	1.176	0.019	0.663	2.186	3.815	~2.90
	0.26	0.25	1.743	0.045	4.107	3.095	9.922	~6.20
	0.31	0.30	4.386	0.232	52.435	5.793	39.793	~16.00
	0.32	0.31	7.647	0.628	185.487	8.718	78.748	~23.00
0.03	0.04	0.01	1.003	0.034	0.011	1.632	1.678	~1.66
	0.13	0.10	1.063	0.046	0.220	1.951	2.571	~2.21
	0.20	0.17	1.222	0.069	0.792	2.389	4.312	~3.33
	0.28	0.25	1.987	0.189	6.052	3.672	12.681	~8.33
	0.31	0.28	3.401	0.505	27.108	5.453	28.961	~16.00
	0.32	0.29	4.845	0.961	63.802	7.113	45.905	~23.00
0.065	0.075	0.01	1.013	0.085	0.013	1.800	1.878	~1.82
	0.165	0.10	1.104	0.124	0.282	2.234	3.054	~2.70
	0.235	0.17	1.338	0.208	1.159	2.900	5.595	~4.33
	0.315	0.25	2.874	0.990	16.240	5.801	23.101	~18.00
	0.325	0.26	3.719	1.619	31.505	7.200	33.080	~26.00

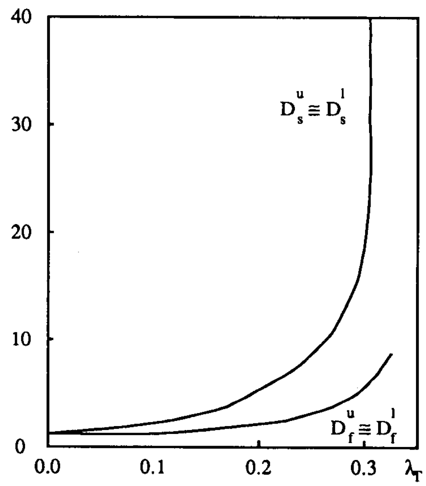

Fig. 2. Bounds on the mean packet delay of the high (D_f^u, D_f^l) and the low (D_s^u, D_s^l) priority classes (in packet lengths) versus λ_T, for $\lambda_f = .01$.

$$\omega_{0,0}^s = \omega_{1,0}^s = 0, \qquad \omega_{0,1}^s = 1;$$
$$\omega_{\mu,0}^s = \omega_{\phi_1+F_1,S_1}^s + \omega_{\mu-\phi_1+F_2,S_2}^s, \quad \mu \geq 2 \tag{4c}$$

$$\omega_{0,\nu}^s = \nu + \omega_{F_1,S_1}^s + \nu\tau_{F_1,S_1} + \omega_{F_2,\sigma_1+S_2}^s + (\nu-\sigma_1)\tau_{F_2,\sigma_1+S_2} + \omega_{F_3,\nu-\sigma_1+S_3}^s, \quad \nu \geq 2 \tag{4d}$$

$$\omega_{\mu,\nu}^s = \nu + \omega_{\phi_1+F_1,S_1}^s + \nu\tau_{\phi_1+F_1,S_1} + (\nu-\sigma_1)\tau_{\mu-\phi_1+F_2,\sigma_1+S_2} + \omega_{\mu-\phi_1+F_2,\sigma_1+S_2}^s + \omega_{F_3,\nu-\sigma_1+S_3}^s, \quad \mu \geq 1, \quad \nu \geq 1 \tag{4e}$$

where all variables are as defined in (1). The bounds on W_f and W_s are obtained for input rates in S_{op} where $S_{op} = \{0 \leq \lambda_f \leq 0.065, 0 \leq \lambda_s \leq \lambda_{s,\max}(\lambda_f)\}$, similarly to those obtained for L. The bounds on L, W_f, W_s for some values of $(\lambda_f, \lambda_s) \in S_{op}$ are shown in Table I.

A similar algorithm for a single class of users which is based on a 2-cell splitting of all packets involved in a conflict (without considering different cells for the different classes) and uses binary feedback information has been found to achieve a maximum stable throughput of $\sim$ 0.36, [4], and induces the mean packet delay D^* shown in Table I. The proposed algorithm for a nonhomogeneous population achieves a total throughput of, at least, between $0.320-0.357$, depending on the contribution of the two classes to the total input traffic. Plots of the bounds on D_f and D_s versus $\lambda_T = \lambda_f + \lambda_s$ are shown in Figs. 2–4.

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 consequence is the increased

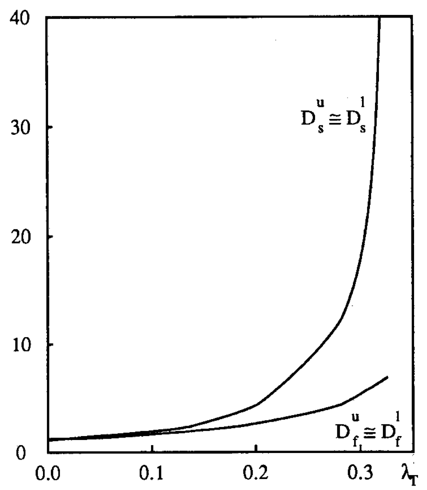

Fig. 3. Bounds on the mean packet delay of the high (D_f^u, D_f^l) and the low (D_s^u, D_s^l) priority classes (in packet lengths) versus λ_T, for $\lambda_f = .03$.

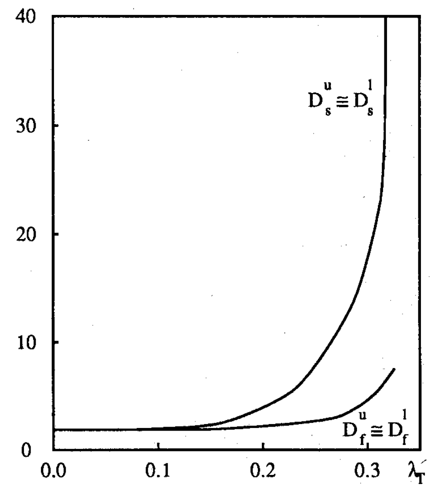

Fig. 4. Bounds on the mean packet delay of the high (D_f^u, D_f^l) and the low (D_s^u, S_s^l) priority classes (in packet lengths) versus λ_T, for $\lambda_f = .065$.

average low priority packet delay compared to the one induced by the homogeneous user population equivalent algorithm. From Table I it is observed that $D_s > D^*$, as expected. On the other hand, a larger amount of packet traffic can be supported by the network if the less time sensitive packet traffic is given lower priority. For instance, if $\lambda_f = 0.065$ packets/slot and the desired average delay is 5.5 to 6.0 slots then (see Table I) an additional traffic of $\lambda_s = 0.185$ packets/slot under the same priority or an additional traffic of $\lambda_s = 0.25$ packets/slot under low priority can be supported. Thus, there is an increase by $\approx 35\%$ of the additional traffic that can be supported if the user population is divided into two classes with different priorities.

References

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

[2] I. Stavrakakis and D. Kazakos, "A multi-user random-access communication system for users with different priorities," Univ. Virginia, Charlottesville, VA, Tech. Rep., UVA/525656/EE87/102, 1987.

[3] ——, "A limited sensing protocol for multiuser packet radio systems," *IEEE Trans. Commun.*, vol. 37, Apr. 1989.

[4] P. Mathys, "Analysis of random access algorithms," Ph.D. dissertation, ETH 7713, Swiss Fed. Inst. Technol. (ETH), Zurich, Switzerland, 1984.