On the Performance of Slotted ALOHA in a Spread Spectrum Environment

Panos Economopoulos
Mart L. Molle

Department of Computer Science
University of Toronto
Toronto, Canada, M5S 1A4

ABSTRACT

We present an extension of the slotted ALOHA protocol for use in a spread spectrum packet radio environment. With spread spectrum, we assume that N distinct codes are available, and that each code can be treated as a separate channel. Running an independent copy of the protocol on each of these channels would be undesirable, since each user would have to select one channel to monitor for packets addressed to it, inducing a logical partitioning of the user population into N groups. To preserve the logical connectivity of the network, we examine the effect of separating packet transmissions into two parts, a short *preamble*, which is sent over a public channel, and followed by the *body*, which is sent over a private channel. We assume that m of the available codes are used as preamble channels, and the remaining $N - m$ codes are used for actual packet transmissions. If $m \ll N$, then the network can support broadcast and multicast packets, and still make use of all of the available channels.

1. Introduction

As the name suggests, packet transmissions in a spread spectrum system[1,2] are dispersed over a wide frequency band rather than concentrated into a narrow frequency band as is usually the case. In return for this apparent waste of channel bandwidth, spread spectrum can be used to reduce the channel error probability for noisy channels, and to improve the security of transmissions in a hostile environment by making it more difficult for an adversary to jam the channel. It may even be possible to hide the fact that transmissions are taking place from an adversary who is not told the details of the spread spectrum modulation technique being used. We note that with spread spectrum, this same wide frequency band can support more than one simultaneous transmission using different codes. If a frequency band that is μ times wider than necessary is used, then roughly μ concurrent packet transmissions can be accomodated on the channel using spread spectrum. Thus, from a protocol modelling point of view, the major effect of adopting spread spectrum modulation techniques seems to be a partitioning of the physical channel into N separate subchannels, each with $1/N$ th of the original bandwidth.

A disadvantage of spread spectrum over an equivalent single channel system is that its partitioning of the channel can lead to a partitioning of the users. We assume that hardware is not duplicated at each user so that a user must first decide to monitor a particular code before it can receive any transmissions that were sent using that code. Thus to contact a particular user, one must be able to find out which code is monitored by that user. In principle, one can always keep track of this information somehow. However, if the same packet must be sent to several users, this partitioning can cause problems. In a single channel system, only one copy of the packet addressed to several users would have to be transmitted if the users understood broadcast (or multicast) addressing. Even with such an addressing scheme, duplicate copies of the packet would have to be transmitted using different codes in the multiple channel system, unless the entire set of recipients happened to monitor the same code.

One idea for improving the logical connectivity of a spread spectrum system with multiple codes is to assign some of the codes to preamble channels, where packet transmissions can be initiated in a manner that is highly visible to all the users, and the remainder of the codes to message channels, where the major part of each transmission can take place in private[3] In this paper, we evaluate this idea when the slotted ALOHA protocol is used to control access to the preamble channels.

2. The Model

Consider a network where an infinite number of identical users transmit constant length packets using a set of N independent channels, each with the same data rate. Without loss of generality, we assume that the transmission time for a packet would be unity if we could make use of the full bandwidth of all the channels, and that the worst-case propagation time between users is α. However, since each transmission uses only one channel, the actual packet transmission time will be N time units.

Recall that in the proposal we wish to study, a preamble is sent on one channel and the remainder of the packet is sent on another. We will consider the preamble to be additional overhead of duration r times the original packet length. Thus, accounting for the propagation time, the slot size must be at least $rN + \alpha$ on the preamble channel, and $N + \alpha$ on the message channel. For simplicity we let each message channel slot be the same length as L preamble channel slots (by adding some dead time to the end of each message slot if necessary), where

$$L = \left\lceil \frac{N + \alpha}{rN + \alpha} \right\rceil \approx \frac{1}{r}. \tag{1}$$

Thus, $T_p = rN + \alpha$ is the elementary time unit in our model.

corresponding packet transmission will take place on a message channel belonging to the $t \bmod L$ th group, starting in the $t+1$st slot. We note that

$$N = L \cdot k + m, \tag{2}$$

and that we may as well assume that $k \leq m$ since it is impossible for our proposed protocol to use more message channels concurrently.

The model of protocol operation is as follows. The total traffic obeys the "strong" Poisson assumption, which asserts that the union of the arrival times for new packets and the retransmission times for collided packets forms a Poisson process with intensity G per unit time. Thus, assuming a uniform distribution of the traffic over the available preamble channels, the number of transmissions in any preamble slot will have a Poisson distribution with parameter

$$G_p = \frac{GT_p}{m}, \tag{3}$$

and the probability of a successful transmission occurring in a preamble slot, p, is given by the well-known formula for slotted ALOHA:

$$p = G_p \, e^{-G_p} \leq \frac{1}{e}. \tag{4}$$

If its preamble is not transmitted successfully (i.e., it is involved in a collision on the preamble channel), then the user waits for a random retransmission delay

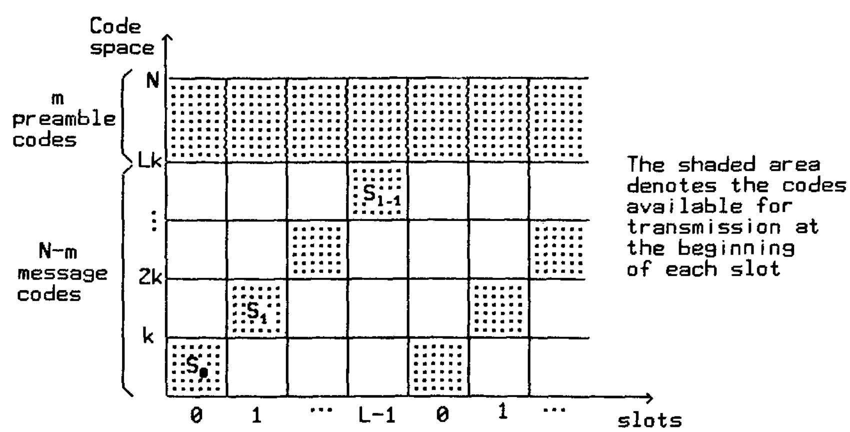
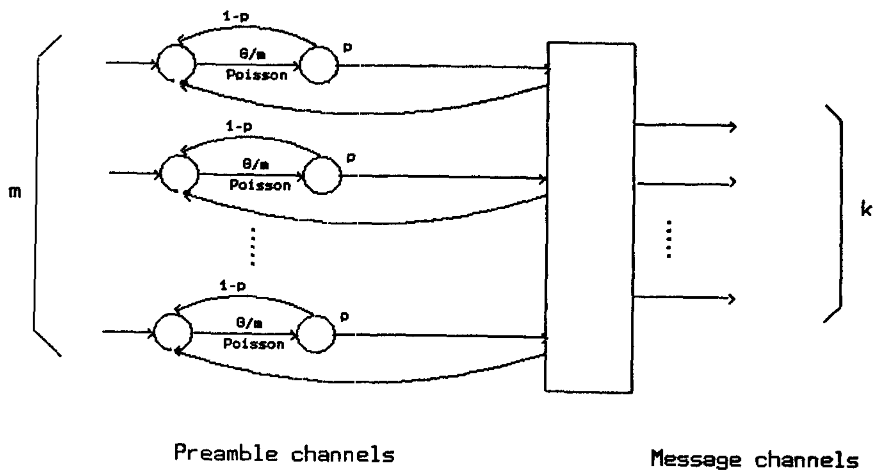


Figure 1

In Figure 1 we show how the N available channels in our model are used: $m \geq 1$ of the channels are used to send preambles, each one slot long, and the remaining $N - m$ channels used to send packets, each L slots long. The message channels are divided into L groups of k channels each, such that the starting points for packet transmissions vary between the groups. If a preamble is sent successfully in the t th slot, then the

and tries again. Otherwise the preamble is assumed to have been transmitted successfully; the user selects one of the available message channels and transmits its packet. If that packet transmission is not successful (because of a collision on the message channel), then the user waits for a random retransmission delay and tries to sent its preamble again. Otherwise the packet transmission is assumed to be successful.



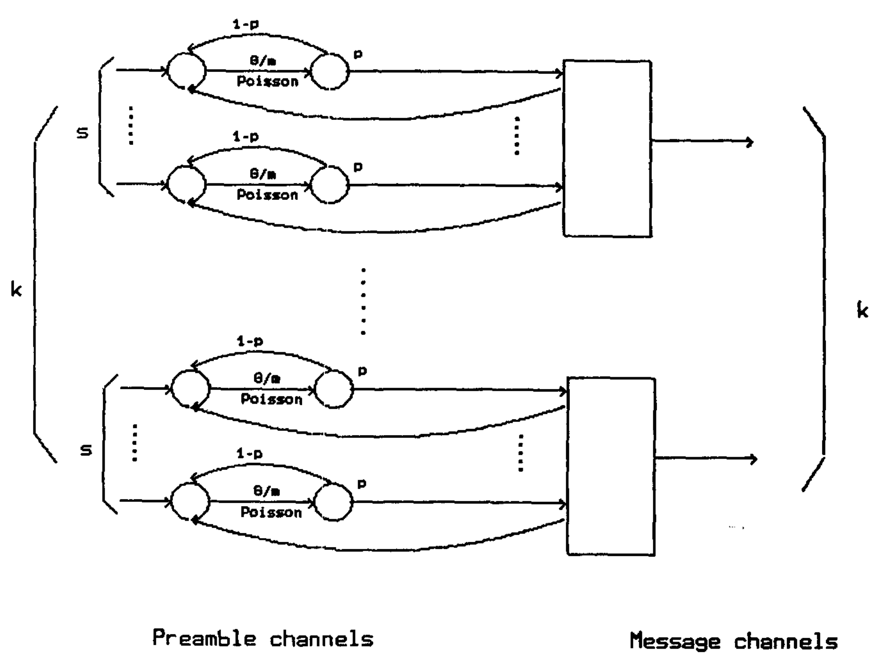
Operation of protocol P1
Figure 2

Note that the operation of the protocol can be decomposed into L subsystems. Each subsystem controls k dedicated message channels and shares the m preamble channels with the other $L-1$ subsystems using round-robin TDMA. However, because we have assumed that the number of users is infinite and that the total traffic is Poisson, the analysis for each subsystem can be done independently.

3. Analysis

At this point, we must decide on the manner in which a user that has sent its preamble successfully is allowed to select a message channel. We consider two cases.

In protocol **P1** (Figure 2), users randomly select one of the k available message channels with equal probability.



Operation of protocol P2
Figure 3

In protocol **P2** (Figure 3), users deterministically select one of the k message channels based on the preamble channel that was used. (Thus in protocol **P2**, each of the k available message channels at each slot has assigned to it the load from $s = \frac{m}{k}$ of the preamble channels.)

3.1. Throughput

To find the throughput equations, we must find what proportion of the time-bandwidth product occupied by each subsystem is used to carry successful packet transmissions.

In protocol **P1**, the average number of successfully transmitted messages is $mp\,(1-\frac{p}{k})^{m-1}$, since a message will succeed if it has a successful preamble and at the same time no other user with a successful preamble on one of the remaining $m-1$ preamble channels chooses the same message channel. The probability of a success on the preamble slot, p, is given in (4). The corresponding channel resource used is $k\,(1+\frac{a}{N})+m\,(r+\frac{a}{N})$ (in units of message transmission time). Therefore, the throughput is:

$$S_1 = \frac{mp\,(1-\frac{p}{k})^{m-1}}{k\,(1+\frac{a}{N})+m\,(r+\frac{a}{N})} = \frac{sp\,(1-\frac{p}{k})^{m-1}}{1+sr+(s+1)\frac{a}{N}} \quad (5)$$

In order to find the throughput in protocol **P2**, let us take a look at the subsystem of a set of s preambles and the corresponding message channel. The probability of a successful transmission on the message channel is $sp\,(1-p)^{s-1}$, since we must have only one success on a preamble channel among the s participating preamble channels. p is still given by (4). The time period considered is 1 message slot and s preamble slots. Therefore, throughput is:

$$S_2 = \frac{sp\,(1-p)^{s-1}}{(1+\frac{a}{N})+s\,(r+\frac{a}{N})} = \frac{sp\,(1-p)^{s-1}}{1+sr+(s+1)\frac{a}{N}} \quad (6)$$

3.2. Comparison of P1 and P2.

In order to compare the two protocols, it suffices to compare the quantities

$$f\,(1) = m\frac{p}{k}(1-\frac{p}{k})^{m-1} \quad \text{and}$$

$$g\,(1) = \frac{m}{k}p\,(1-p)^{\frac{m}{k}-1} \quad (7)$$

where

$$f\,(j) = \mathbf{P}[j \text{ transmissions in a message slot given } \mathbf{P1}]$$

$$= \binom{m}{j}\left(\frac{p}{k}\right)^j\left(1-\frac{p}{k}\right)^{m-j} \quad \text{and}$$

$$g\,(j) = \mathbf{P}[j \text{ transmissions in a message slot given } \mathbf{P2}]$$

$$= \binom{m/k}{j}p^j\left(1-p\right)^{\frac{m}{k}-j} \quad (8)$$

Furthermore,

$$f\,(j+1) = f\,(j)\frac{(m-j)p}{(j+1)(k-p)} \quad \text{and}$$

$$g\,(j+1) = g\,(j)\frac{(m-jk)p}{(j+1)k\,(1-p)} \quad (9)$$

Supposing $p \le \frac{k}{m}$, which is the necessary condition so as not to overload the system, $\frac{mp}{k} \le 1$ and clearly,

$$f\,(j+1) \le f\,(j),\ g\,(j+1) \le g\,(j) \ \text{ for } j = 1,2,\cdot \quad (10)$$

We will, now, show that **P2** is better than **P1**, that is:

$$g\,(1) \ge f\,(1) \quad (11)$$

Suppose that (11) is false and $g\,(1) < f\,(1)$. Since

$$\frac{(m-j)p}{(j+1)(k-p)} \le \frac{(m-jk)p}{(j+1)k\,(1-p)} \quad (12)$$

is clearly true and $g\,(1) < f\,(1)$ holds by the assumption, then $g\,(2) < f\,(2)$, $g\,(3) < f\,(3)$, $\cdots$ and hence $E[g\,(\cdot)] < E[f\,(\cdot)]$ which contradicts the fact that

$$E[g\,(\cdot)] = E[f\,(\cdot)] = \frac{mp}{k} = sp. \quad (13)$$

Therefore (11) must hold and protocol **P2** is better than **P1**. This is intuitively expected, since **P2** reduces the contention among messages that have succeeded in the preamble channel. In this case, **P2** behaves as an ALOHA system with a finite population of s "users". Each "user" transmits with probability p, the throughput of the preamble channel. The total traffic of this imaginary ALOHA system is sp, and the throughput will be maximized for $sp=1$, that is, $p=\frac{1}{s}$. Since **P2** gives the best throughput, for reasonable values of p, we will choose protocol **P2** and further analyze its performance. In the subsequent discussion, the throughput will be denoted by S.

3.3. Delay

To model the delay, we use the well-known approximate formula[4, 5]

$$D = D_{pre} + D_{msg}$$

$$= \frac{T_p}{2} + T_p + \left(\frac{G}{S}-1\right)\left(\frac{T_p}{2} + T_p + \bar{R}\right) + N + a$$

where $\bar{R}$ is the average retransmission time of a preamble packet. Note that the quantity $\frac{G}{S}-1$ gives the total number of times a message is retransmitted because of collisions on either the preamble or the message channel. A crude but useful approximation[6] is $\bar{R} = \frac{G}{S}(slots) = \frac{G}{S}\,T_p$. Therefore, finally

$$D = \frac{3}{2}\,T_p + \left(\frac{G}{S}-1\right)\left(\frac{3}{2}+\frac{G}{S}\right)T_p + N + a \quad (14)$$

3.4. System capacity with protocol P2.

It is obvious that the throughput increases as $r \to 0$ and $a \to 0$. Let us now find the capacity of the protocol in the idealized case with $r \to 0$, $N \to \infty$ and $rN \gg a$ or $a \to 0$. Here, the overhead due to the preamble channel is negligible, so $S = sp(1-p)^{s-1}$ and the optimization problem becomes:

$$\begin{array}{ll}\text{maximize} & S = sp(1-p)^{s-1} \\ \text{subject to} & 0 \le p \le e^{-1},\ s \ge 1\end{array} \qquad (15)$$

By applying the Lagrange multipliers method, we find that the above maximum occurs at $s = s^* = \frac{1}{1-\ln(e-1)} = 2.18 < e$, for $p = e^{-1}$ and the capacity is

$$C = s^* e^{-1}(1-e^{-1})^{s^*-1} = 0.4667 \qquad (16)$$

This maximum value is attained for $G = G_{max}$, the solution to equation (4) evaluated at $p = e^{-1}$, namely

$$G = \frac{m}{T_p} = \frac{1}{(r+\frac{1}{s})(1+\frac{a}{rN})} \qquad (17)$$

For the given values, we find $G_{max} = s^* = 2.18$.

For $r, a \ne 0$ and N finite, we can find the capacity as a function of s, maximized over p. This function corresponds to the envelopes of the actual $S-G$ curves, for various parameter values. Representative diagrams of S_{max} and G_{max} as functions of s are plotted in Figures 4 and 5 respectively. Of course, these diagrams are dependent on the parameters N, r and a as well. From these plots, one can deduce the most efficient operating point, for a given traffic and channel configuration. Obviously, this must occur at an extreme point or at a place where $\frac{\partial S}{\partial p} = 0$.

Differentiating, $\frac{\partial S}{\partial p} = \frac{s(1-p)^{s-2}(1-sp)}{1+sr+(s+1)\frac{a}{N}}$. We consider two cases. First, for $1 \le s < e$ we find that $\frac{\partial S}{\partial p} > 0$ since $0 \le p \le e^{-1}$. This establishes that $S_{(1 \le s < e)}$ is monotonically increasing and, consequently, that

$$S_{max,(1 \le s < e)} = \frac{se^{-1}(1-e^{-1})^{s-1}}{1+sr+(s+1)\frac{a}{N}} \quad \text{for}$$

$$G = \frac{1}{\frac{1}{s}+r+\frac{(r+1)a}{rN}} \qquad (18)$$

For $s = 1$,

$$S_{max,(s=1)} = \frac{e^{-1}}{1+r+\frac{2a}{N}}, \quad \text{for}$$

$$G = \frac{1}{1+r+\frac{(r+1)a}{rN}} \qquad (19)$$

Obviously, in this case, we have m independent ALOHA subsystems, each one consisting of a preamble and a message channel. The denominator accounts for the loss of useful capacity, due to the preamble channel.

For the second case we consider $s > e$. Here, S has a maximum at $p = \frac{1}{s}$, for which $\frac{\partial S}{\partial p} = 0$ and

$$S_{max(s>e)} = \frac{s\frac{1}{s}(1-\frac{1}{s})^{s-1}}{1+sr+(s+1)\frac{a}{N}} = \frac{(1-\frac{1}{s})^{s-1}}{1+sr+(s+1)\frac{a}{N}} \qquad (20)$$

Note that there are two values of G that satisfy the corresponding equation (4) with $p = \frac{1}{s}$, therefore, the fork-like diagram of Figure 4.

The above analysis shows that, for small s, it pays to saturate the preamble channels so as to attain S_{max}. As s becomes larger, too many messages pass successfully through the preamble channels and interact destructively over the message channel. In that case, the optimal is obtained by *decreasing* the traffic at the preamble channels and keeping at saturation the traffic of the message channel.

4. Studying the protocol numerically

We shall now study the effect of changing the system parameters, like r, s, N and a on the S and D. It is not possible to study the effect that each of them has on the system throughput analytically. Our comments will be based on the formulas developed so far and the corresponding diagrams.

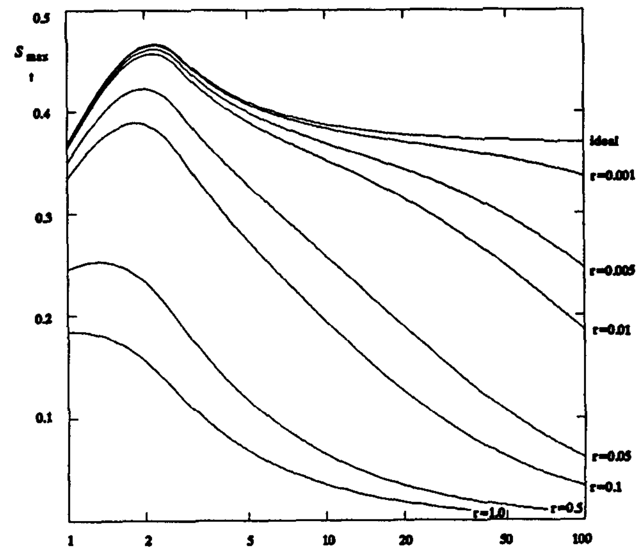
Varying s, the number of preamble channels assigned to each message channel : From the S_{max} vs. s curve (Figure 4) and the S vs. G curve (Figure 6a,b) we can deduce that S decreases with increasing s. Furthermore, for increasing s, the $S-G$ curves shift to the left, which means that the system gets saturated at smaller values of G.

Varying r, the ratio of a preamble length to a message length : The throughput decreases monotonically with increasing r (Figure 7a,b). This is intuitively expected, since the preamble channel does not contribute to the throughput. Again the $S-G$ curves shift to the left for increasing r.

Varying a, the worst-case propagation time and N, the number of the available sub-channels : a and N appear in the formulas only as $\frac{a}{N}$, which is the propagation delay relative to the actual message length. Obviously, for $\frac{a}{N} \ll 1$ the throughput formula is quite insensitive to the exact values of a and N (Figures 8a,b and 9a,b). On the other hand, D is directly proportional to N and, therefore, increases with increasing N.

5. Conclusions

We have shown that it is feasible to modify the slotted ALOHA protocol so that each user must first transmit a short preamble successfully on a "public" preamble channel before he is allowed to transmit his packet on a "private" message channel. This allows us to enhance the logical connectivity of a multi-channel spread spectrum packet radio network, and still make use of all the available channels efficiently. In fact, when the preamble is small, our modified slotted ALOHA protocol actually out performs slotted ALOHA because the preamble channel is equivalent to a weak form of collision detection.



S_{max} versus s
Figure 4

References

1. D. C. Dixon, *Spread Spectrum Systems*, Wiley-Interscience, New York (1976).

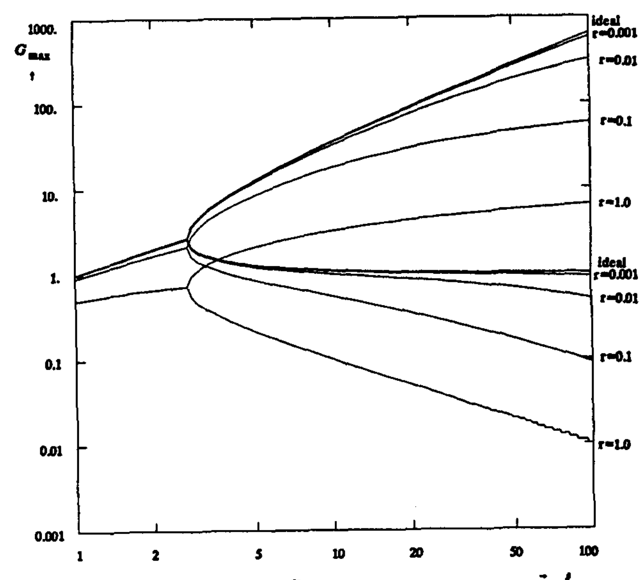
2. R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman, "Advances in Packet Radio Technology," *Proceedings of the IEEE* **66** pp. 1468-1496 (November 1978).

3. *UCLA Packet Radio Analytical Workshop*, UCLA Computer Science Department, (August 10-11, 1982).

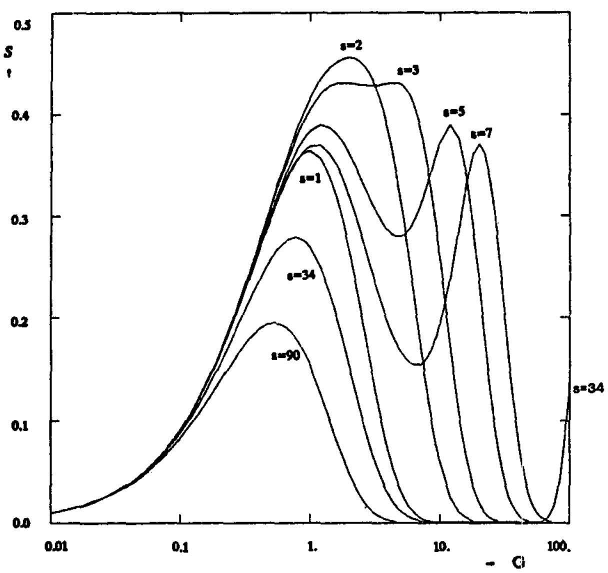
4. L. Kleinrock and F. A. Tobagi, "Packet Switching in Radio Channels: Part I — Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Transactions on Communications* **COM-23**(12) pp. 1400-1416 (December 1975).

5. L. Kleinrock, *Queueing Systems, Vol II., Computer Applications*, Wiley-Interscience, New York (1976).

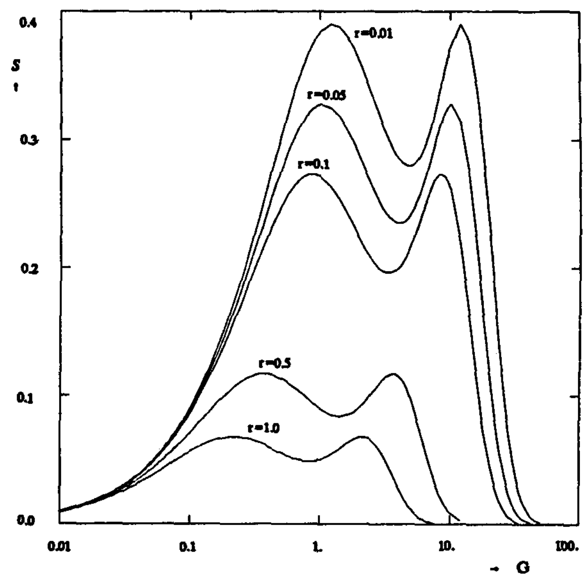
6. M. L. Molle and L. Kleinrock, "Virtual Time CSMA: A New Protocol with Improved Delay Characteristics," CSD Report No. 810113, Computer Science Department, University of California, Los Angeles (January 13, 1981).



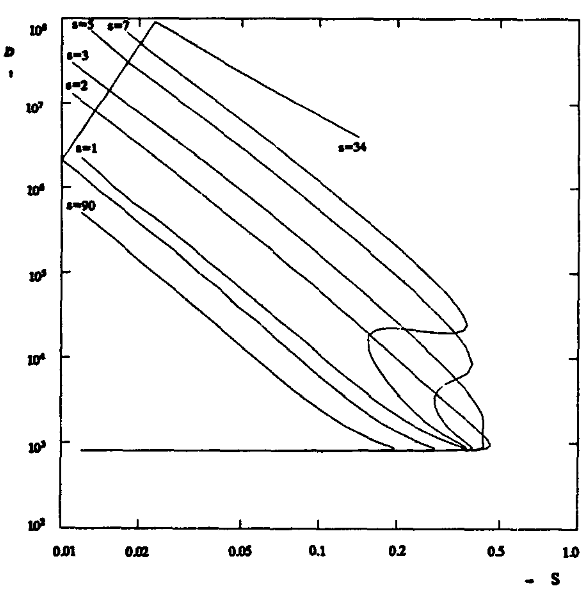
G_{max} versus s
Figure 5



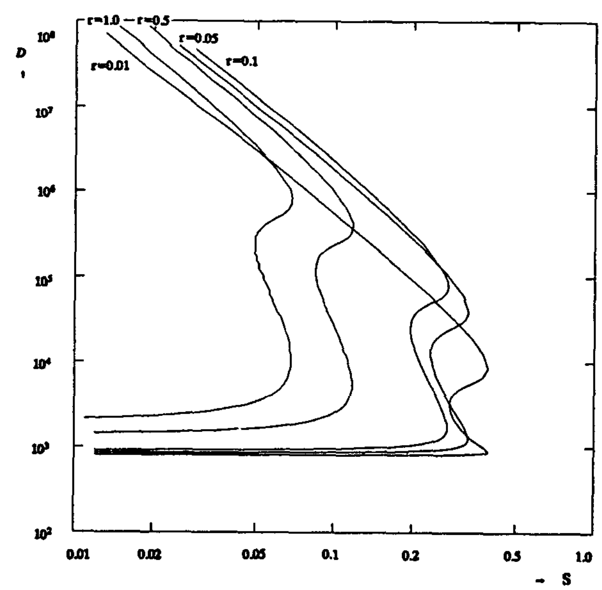
Throughput **versus** ***Traffic*** **diagram for r =0.01 and s varying**
Figure 6a



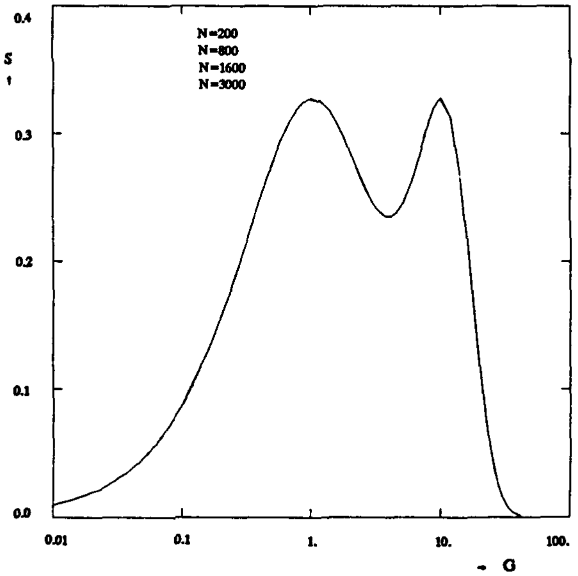
Throughput **versus** ***Traffic*** **diagram for s =5 and r varying**
Figure 7a



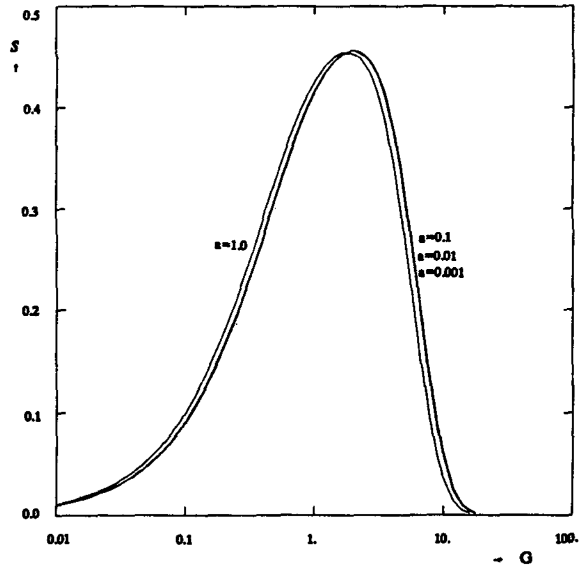
Delay **versus** ***Throughput*** **diagram for r =0.01 and s varying**
Figure 6b



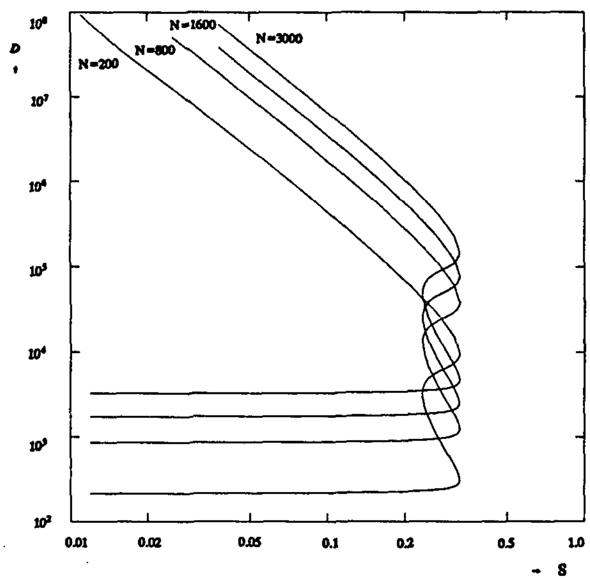
Delay **versus** ***Throughput*** **diagram for s =5 and r varying**
Figure 7b



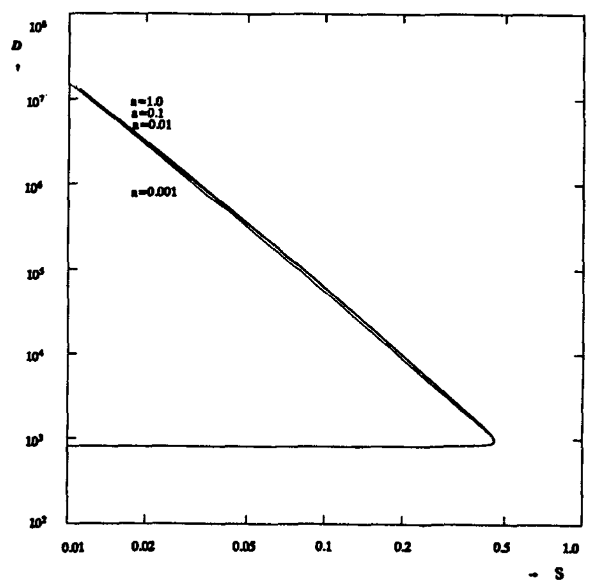
Throughput **versus** ***Traffic*** **diagram**
for *r* =0.05, *s* =5 and *N* varying
Figure 8a



Throughput **versus** ***Traffic*** **diagram**
for *r* =0.01, *s* =2 and *a* varying
Figure 9a



Delay **versus** ***Throughput*** **diagram**
for *r* =0.05, *s* =5 and *N* varying
Figure 8b



Delay **versus** ***Throughput*** **diagram**
for *r* =0.01, *s* =2 and *a* varying
Figure 9b