ARE PRIORITIES USEFUL IN AN 802.5 TOKEN RING?

Jeffery H. Peden & Alfred C. Weaver

Computer Science Report No. TR-88-05
February 23, 1988

This work was sponsored by the Flight Data Systems Branch
of NASA--Johnson Space Center, Houston, Texas.

Are Priorities Useful in an 802.5 Token Ring?

Jeffery H. Peden & Alfred C. Weaver
Department of Computer Science
Thornton Hall
University of Virginia
Charlottesville, Virginia 22903

Abstract — The IEEE 802.5 token ring protocol defines eight packet priorities. The intent is that "high priority" packets should be delivered prior to "low priority" packets. Through a series of simulations we show that this expected behavior occurs under a unique set of circumstances: when there are very few network stations, very short data packets (but still long relative to ring latency), very short token hold times, and very high network load.

In the general case, we found that priorities did not markedly influence packet delivery time. Use of the priority system generally resulted in more overhead and longer average packet delays than when all packets were carried as a single priority. We mathematically describe the features of the protocol operation which are the cause of this increased delay and lack of priority discrimination.

I. Introduction

The priority scheme on an IEEE 802.5 token ring operates as follows: Every packet is assigned a priority between 0 (lowest) and 7 (highest). When a station receives the token, the station transmits any enqueued packets that are at or above the priority of the token, higher priority packets first. Enqueued packets that have a priority less than that of the token must wait until a lower priority token is received by the station.

As packets are repeated by non-transmitting stations, these stations may reserve the priority of the token by setting the reservation bits in the packet header so that they equal the highest priority packet that they have waiting (an attempt to lower the value of the reservation field has no effect). When a station has completed its transmission, it retransmits the token at the greater of the value of 1) the current token priority and 2) the current reservation field. If a station raises the token's priority from X to Y, it is responsible for lowering the token's priority from Y to X if it later receives a priority Y token with no reservations higher than X.

II. General Performance

A. Reference Configuration

In order to establish baseline performance for the token ring, a reference configuration was established consisting of 40 active stations, a station repeater latency of 1 bit time with total propagation delay being equal to the sum of the station latencies, a data rate of 1 megabit/second, constant packet lengths of 32 octets (including framing), and Poisson arrivals. Studies were then made of network performance in both the single priority case and the multiple priority case (in which network load was divided evenly among the eight priorities).

Fig. 1 shows the total packet delay for both the single and multiple priority modes of operation. Note the close proximity of the delays for all priorities in the multiple priority curves. The actual delay values shown in Table 1 indicate that there is no significant or consistent difference between the delay experienced by a high priority packet and that of a low priority packet.

Table 2 gives delays for the single priority case in an otherwise identical network configuration. Note that the delay for the multiple priority case is significantly greater in the medium and upper load ranges than the delay for the single priority case. The delay for the multiple priority case is also greater for low loads, but the differences are only a few microseconds.

The increase in delay when multiple priorities are used is due to the cost (as measured in terms of increased delay) of the priority reservation scheme. It has been shown that the cost of the reservation scheme in terms of added delay is a mean increase of one token cycle time for every *distinct* outstanding priority reservation request. The actual total increased delay seen on the network is normally less than one token cycle time because the mean number of reservation requests in any single token cycle is less than one. This will be discussed further in section III.C.

B. Effects of Parameter Variation

To ensure that the observed delay using multiple priorities was not an artifact of the chosen parameters, simulations were run in which the number of packet priorities was varied from two to seven, and the same effect was seen. We therefore reduced the number of packet priorities to only two, and varied the load distribution between them.

Simulations were run such that the fraction of load given to the low and high priorities was 0.1 and 0.9 respectively, and then varied in increments of 0.1 until the distribution of load between low and high priorities was reversed to 0.9 and 0.1 respectively. The results of these simulations indicate that the number of priorities offered to the ring, as well as the relative fraction of bandwidth assigned to each priority, still results in increased overall delay for all priorities. That is, the number of priorities used does not impact the cost of the reservation scheme in the sense that there is still a token cycle delay for each reservation request.

One case in which the added delay will be virtually unnoticeable occurs when the overwhelming majority of the load is one priority, i.e., packets of other priorities are ''incidental''. In this case, there will be very few distinct reservation requests, resulting in much lower added delay.

It was shown in [2] that for any particular offered load, achieved throughput for the token ring decreases as packet sizes become shorter, and that this loss of throughput is exaggerated when the number of ring stations is small. It was observed that the effects of multiple priority operation only had significant impact when the number of ring stations was less than 40. It was also shown that network efficiency increases as packets become larger; that is, the average delivery delay of a bit within a packet decreases as packet size increases. We therefore dealt with only the ''interesting'' cases in which the number of stations was 40 or less, and where packets were short with respect to ring latency. Our packets were 32 octets (13 of which are MAC layer framing), with a transmission time of 256 μsec; a ring of N stations has a latency of $N + 27$ μsec.

III. Priority Effects

A. General Performance

In this section we present and discuss the actual effects that the priority operation has on delay. Fig. 2 shows that for 20 ring stations packet sizes of 32 octets, and where multiple transmitted packets per received token are allowed (MPPT service), there is minimal difference between the delays experienced by high and low priority packets. Fig. 3 shows the priority delay for a ring configuration of 3 stations, again with 32 octet packets. Note that for 3 stations, where the difference between delays for different priorities is the highest, it is less than a factor of two from highest to lowest, and this does not occur until an offered load of 95% is reached (a loading difficult to achieve with 3 stations).

To get the maximum effect from the operation of the priorities, simulations were run in which the ring was significantly overloaded for several seconds. Fig. 4 shows the results observed. Note that service shutoff does not occur for the lowest priority until an offered load of over 99% is reached. Also note that the priority operation is very well behaved, in that service shutoff occurs in order of increasing priority, and that higher priorities continue to receive guaranteed delay until their service shutoff.

B. Queueing Delay

One interesting effect is that the queueing delay for a packet is, in the mean case, independent of that packet's priority. This effect is independent of whether a priority queue or multiple queues are implemented.

In the general operating case, the token arrives at a station to find either zero or one packet enqueued for transmission. Given that this is the case, this has the result that service of the queue in priority order and service of the queue in FIFO order are identical. Under the assumption of constant service times (an assumption we make) the equation for the queueing delay $E[Q]$ of a packet of any priority is shown in [3] to be

$$E[Q] = \frac{\Lambda \frac{1}{\mu^2}}{2(1-\Phi)}, \quad N\Phi < 1 \tag{1}$$

where

$\Lambda \equiv$ the sum of the arrival rates of packets of all priorities

$\mu \equiv$ the service rate in packets per second

$\Phi \equiv$ the total load contribution of a single station, $\Phi = \frac{\Lambda}{\mu}$

$N \equiv$ the number of ring stations

The limit $N\Phi < 1$ rather than $\Phi < 1$ is necessary since Φ is the total offered load at a single station. We assume this limit throughout our discussion.

C. Priority Reservations

One of the factors increasing the delay when multiple priorities are offered to the network is the reservation mechanism. As tokens and packets circulate on the ring, stations which have enqueued packets below the priority of the token may attempt to reserve the priority of the token. This is done by setting a 3-bit subfield in the access control field of the token or packet. All packets are initially transmitted with a reservation field value of zero, thus allowing stations with lower priority packets to make reservations. Once a station has made a reservation, an attempt to lower the value written has no effect.

The amount of time that a station may transmit is governed by the token holding timer at that station (we assume that all stations have the same value for these timers). The default value for this timer is given in the standard as 10 msec. [1]. Therefore, the expression for the number of packets per token that may be delivered per token reception is given by

$$\xi = \lfloor \Omega\mu \rfloor \tag{2}$$

where

$\Omega \equiv$ the setting of the token holding timer in seconds

The *floor* function is necessary because the standard states that this timer may not be overrun; that is, prior to each packet transmission, this timer is checked to see if there is enough time remaining to complete the pending transmission.

The number of attempted reservations R during one token cycle is given by equation (3). Note that the number of reservations expected may include multiple reservation requests for the same priority.

$$R = N\left(1 - \exp\left(-\Lambda\left(E[V]\xi^{-\Phi(N-1)} + \frac{1}{\mu}\right)\right)\right) \qquad (3)$$

where

$E[V] \equiv$ the mean token vacation time, i.e., the time between a station's transmission of the token and that station's later reception of the same token

The number of *distinct* reservation requests is the total number of different token priorities requested during the token cycle. We define $\mathbf{X}$ to be the vector random variable which takes on the value of the total number of reservation requests made, and $\underline{\mathbf{X}}$ to be the total number of X_i components having non-zero values.

Using the above definitions, the equation for the total delay overhead caused by the reservation mechanism is

$$\sum_{\underline{\mathbf{X}}=1}^{R}\left(\underline{\mathbf{X}} \cdot P(\mathbf{X}=\mathbf{k}) \cdot \left(E[V] + \frac{1}{\mu}\right)\right) \qquad (4)$$

The probability distribution of $\mathbf{X}$ is an ordinary multinomial distribution and is given by

$$P(\mathbf{X}=\mathbf{k}) = \begin{bmatrix} R+1 \\ k_1, k_2, \cdots, k_n \end{bmatrix} \prod_{i=1}^{n} \left[\frac{r_i}{n}\right]^{k_i} \qquad (5)$$

where

$r_i \equiv$ the proportion of the total load of packets of priority i

$n \equiv$ the number of priorities being offered to the network

The reason for the $R + 1$ term instead of R is that the priority of the token counts as a reservation for the purposes of computing the number of distinct reservation requests, since a reservation request for a priority equal to that of the token will have no effect (i.e., there is no added delay).

D. Effective Load

The minimum possible load that can affect the delay of priority i packets is the load contribution of priority i plus the load contribution of all higher priorities. The maximum possible load that can affect the delay of any packet is the total network load. The actual load affecting priority i packets lies between these two limits, and is called the *effective* load. The equation for the effective load σ_i for priority i is

$$\sigma_i = \Phi - \Phi\, \xi^{-\Phi (N-1)} (\Phi - \delta_i) \tag{6}$$

where

$\delta_i \equiv$ the load contribution at a single station of all priorities greater than or equal to priority i

The effective load for priority i decreases in proportion to the load at a station, and increases in proportion to the number of stations as well as the number of packets per token allowed. Note that this is a major reason that the added delay introduced by the reservation mechanism affects *all* priorities, not just the priorities which are lower than the token's priority.

E. Token Hold Times

Up to this point, the effects of the priority scheme have been studied with the ring operating under the condition of the token holding timer being set to the default value. This is a reasonable assumption to make, as it is anticipated that few users of this network will make changes in the default values of the various network parameters defined in the standard.

It was shown in [2] that the default setting of the token holding timer usually results in exhaustive service of the packet queues. This is true even at very high network loads, and is generally independent of packet length. Therefore, simulations were run in which the token holding timer was set to less than the default value, in order to check the effects of the priority scheme under the condition of non-exhaustive service. Again it was seen that as the number of ring stations increases to 40 and above, the effects of the priority operation rapidly become insignificant, even under single-packet-per-token (SPPT) service.

Fig. 5 shows the service delay for 20 stations when the token holding timer has been set such that SPPT service results. Packets are 32 octets in length. Note that when the token holding timer is set such that SPPT service results, $\xi = 1$, which reduces equation (3) to

$$R = N\,(1 - \exp(-\Lambda\,(E\,[V] + \frac{1}{\mu}))) \tag{7}$$

thus increasing the expected number of reservation attempts. Setting $\xi = 1$ also reduces equation (6) to

$$\sigma_i = \Phi - \Phi\,(\Phi - \delta_i) \tag{8}$$

which has the effect of decreasing the effective load for high priority packets, resulting in much greater discrimination between the priorities.

The increasing effect of the priority scheme can be seen in Fig. 6. Note that the graphs do not show delays for low loads. This is because there is negligible difference at low loads between the delays of the various priorities. One interesting effect is that as the number of stations decreases, the delay of the lowest priority is greater than that of a network configuration with a larger number of stations, and the delay of the highest priority is less. This is another indication that the priority scheme is operating as intended, i.e., lower priorities are taking the brunt of the increasing delay.

IV. Conclusions

It is evident that unless very high network loads are expected in the ordinary course of events, the network performs better without the added overhead and complexity of the priority operation. The greatest effects of the priority scheme were obtained with very short packets, a low number of network stations, and SPPT service. The only way to limit transmission to one packet per token is to change the token holding timer such that there is only time for the transmission of one packet when the token is received. However, this will tend to reduce the token holding timer to the point where only very short packets are allowed (if only large packets are transmitted, this in itself reduces the effect of the priority scheme). It is also a difficult matter to generate a high network load with a small number of stations.

We conclude that, in general, the priority mechanism is not a very useful feature of the 802.5 protocol. While there are some circumstances in which priority operation is useful, they are rare. Given the added delay and overhead generated under ordinary conditions by the use of the priority scheme, and the small set of circumstances under which multiple priorities are shown to be useful, we have shown that under general conditions, network traffic should be offered at a single priority, and that this will result in better service (lower delays) the majority of the time.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of NASA-Johnson Space Center, NASA-Lewis Research Center, Sperry Marine Inc., and the Institute of Information Technology of the Virginia Center for Innovative Technology.

REFERENCES

[1] *Token Ring Access Method and Physical Layer Specifications*, The Institute of Electrical and Electronics Engineers, Inc., 1985.

[2] J. H. Peden, *Performance Analysis of the IEEE 802.5 Token Ring*, Master's Thesis, Department of Computer Science, University of Virginia, January 1987.

[3] M. Schwartz, *Telecommunications Networks: Protocols, Modeling and Analysis*, Addison-Wesley, 1987.

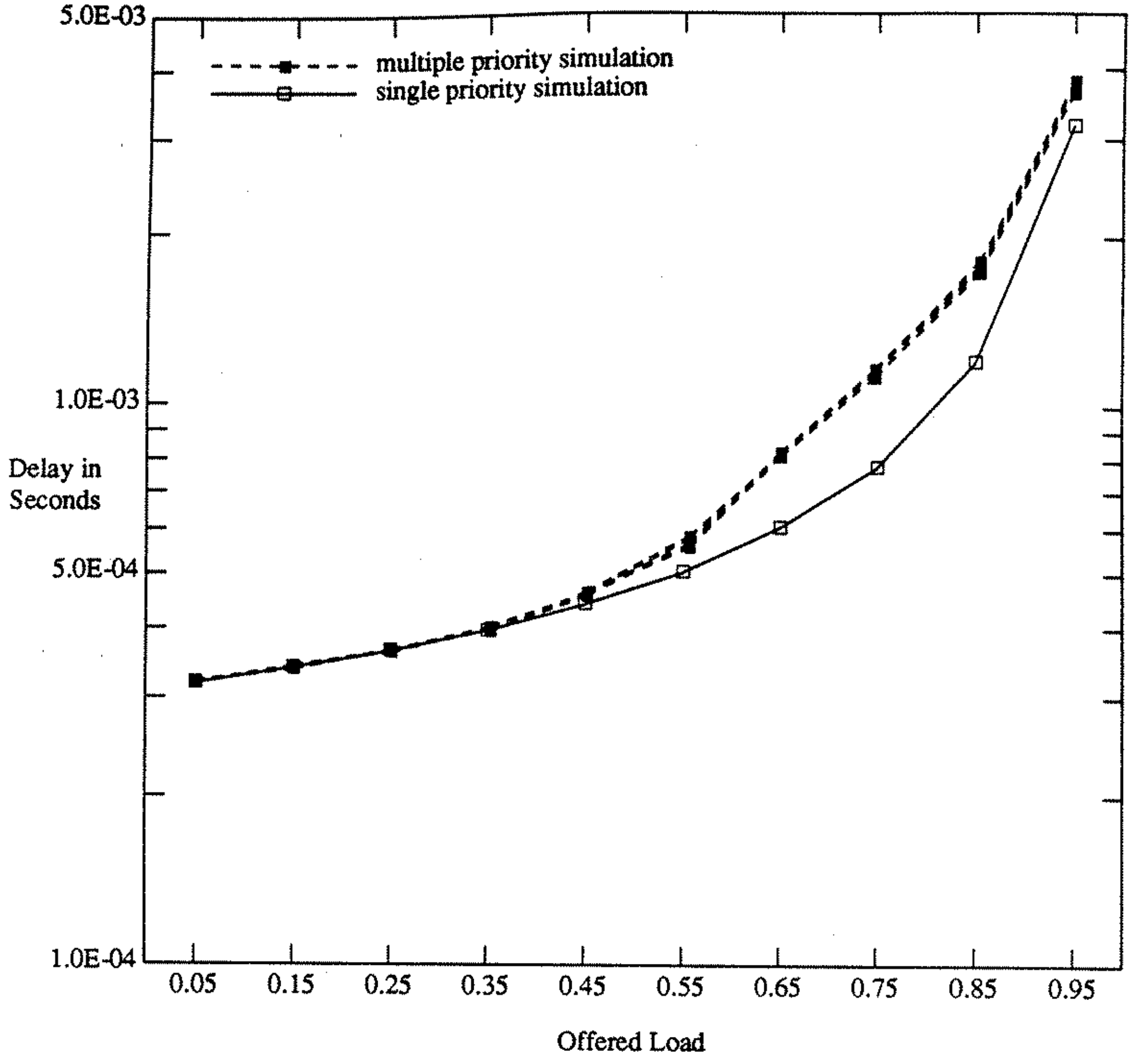

Figure 1
Delay for single and multiple priority operation, 40 stations.

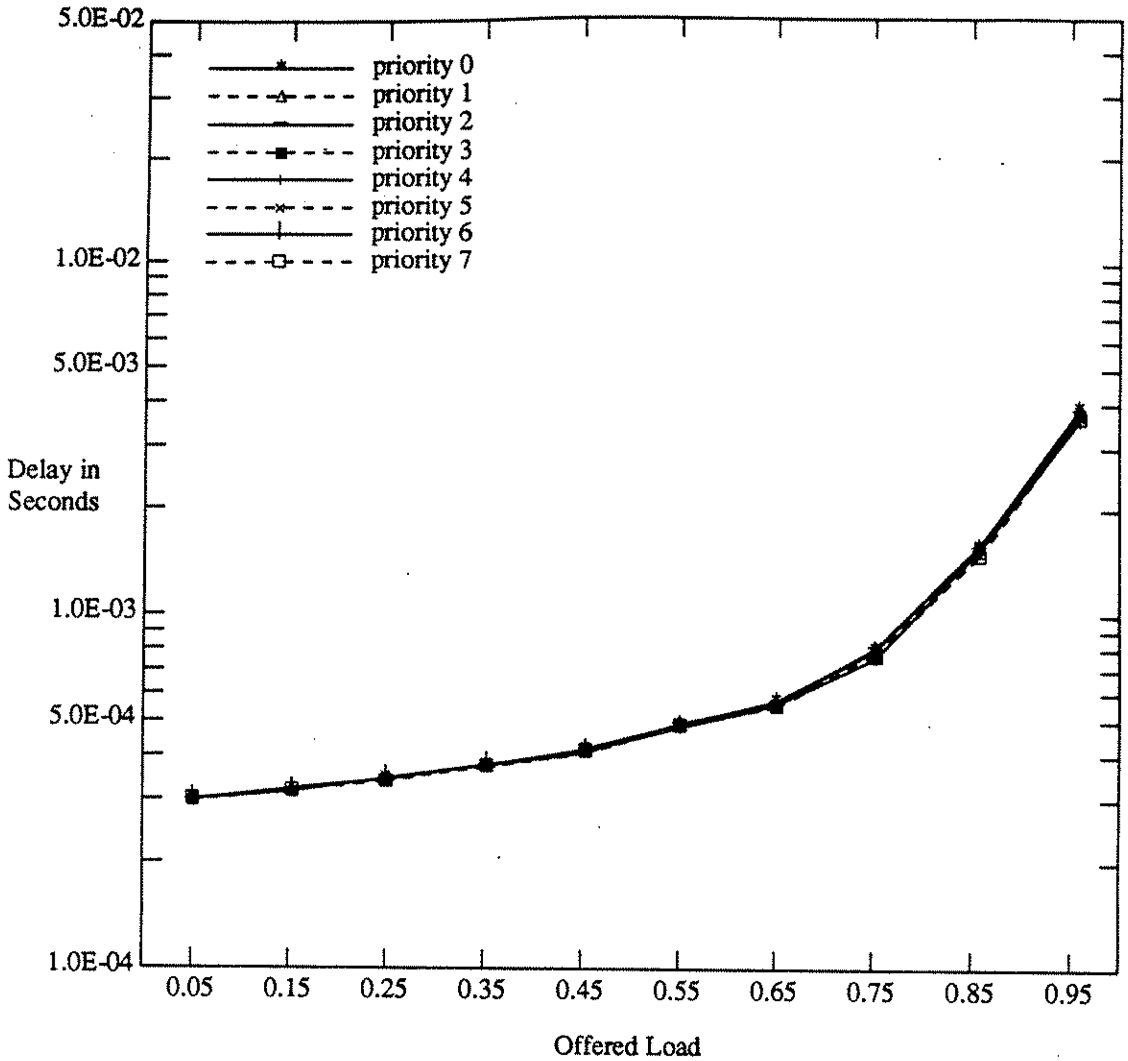

Figure 2
Delay for the MPPT service mode, 20 stations.

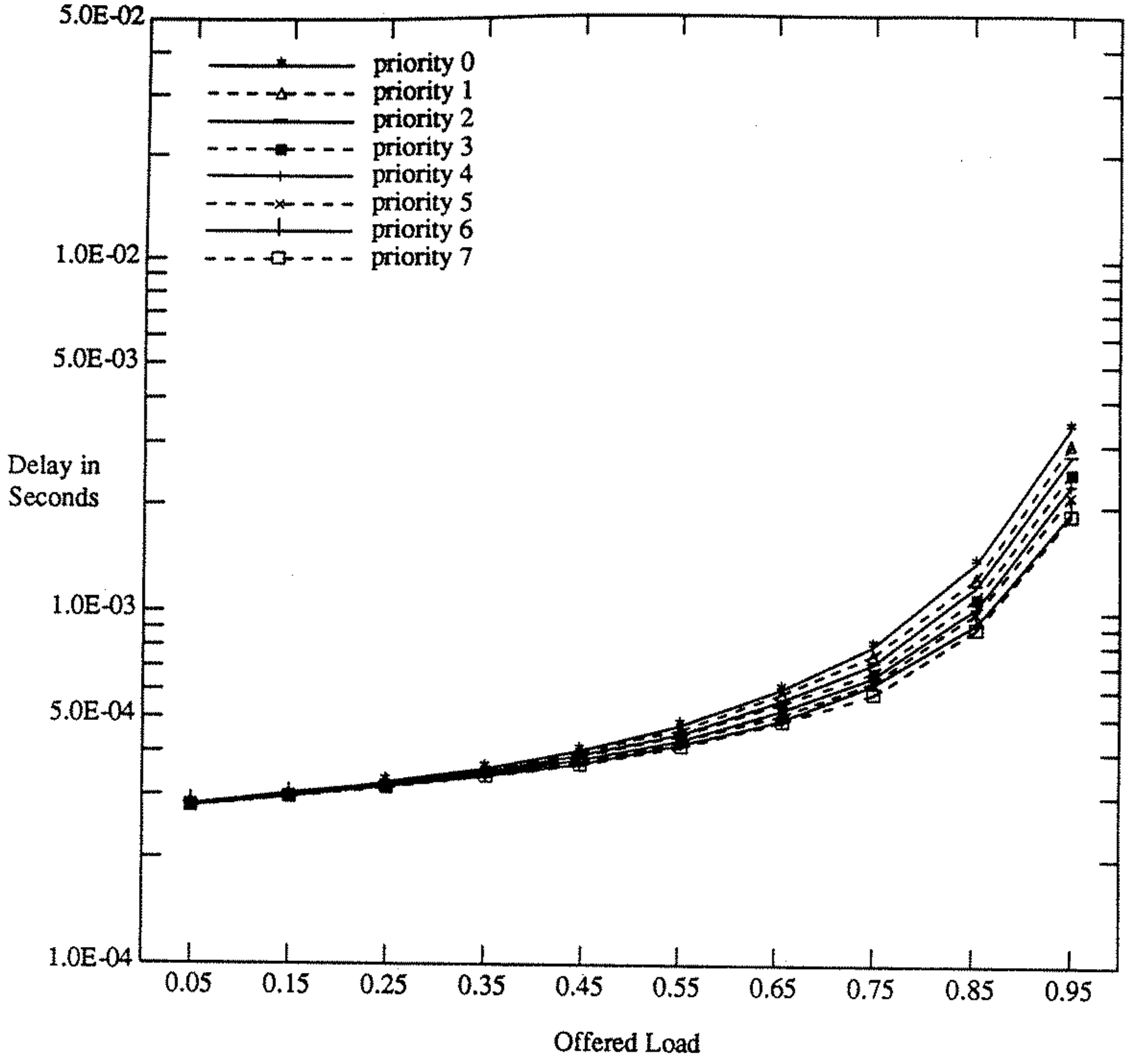

Figure 3
Delay for the MPPT service mode, 3 stations.

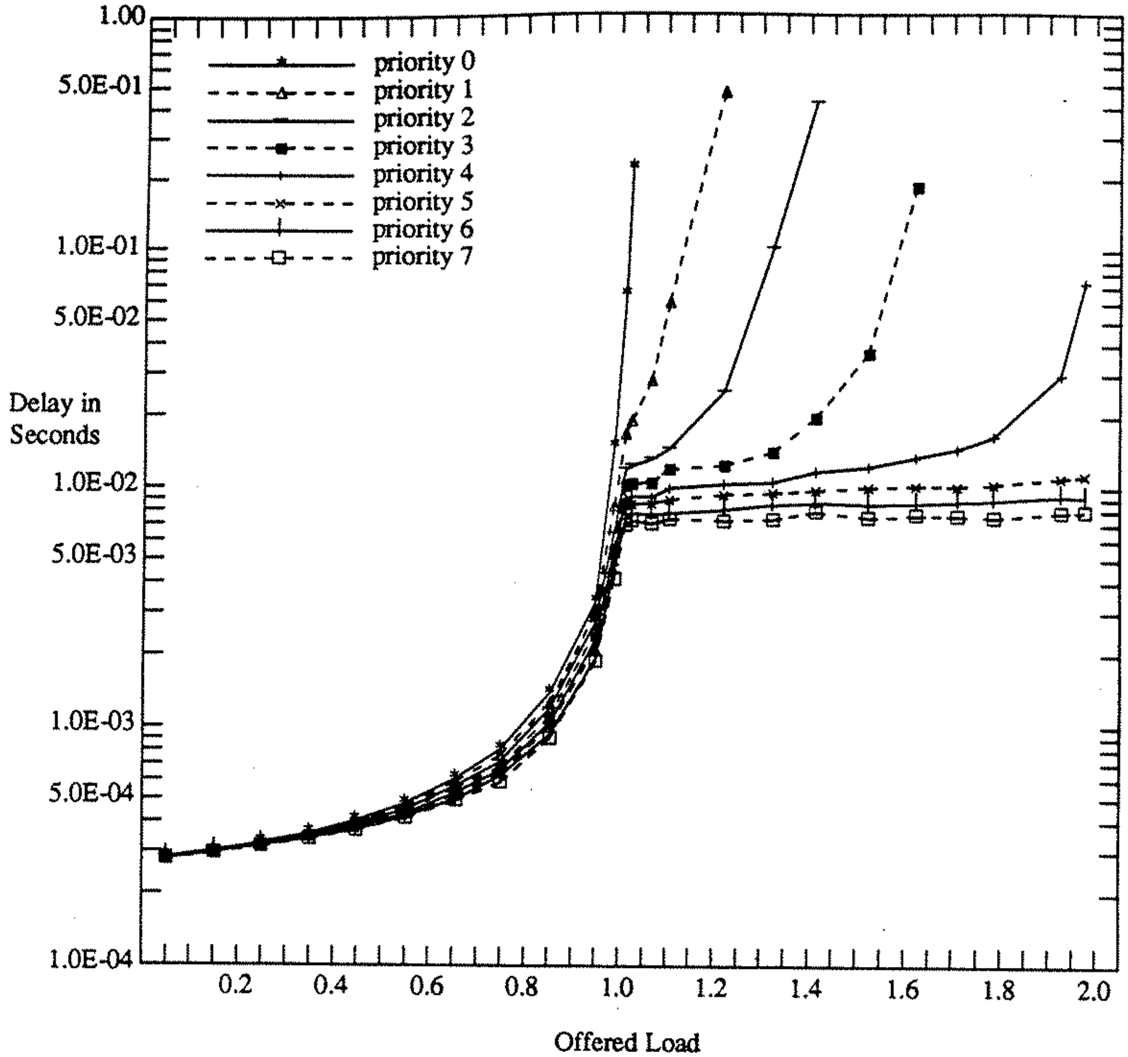

Figure 4
Service shutoff for the MPPT service mode, 3 stations.

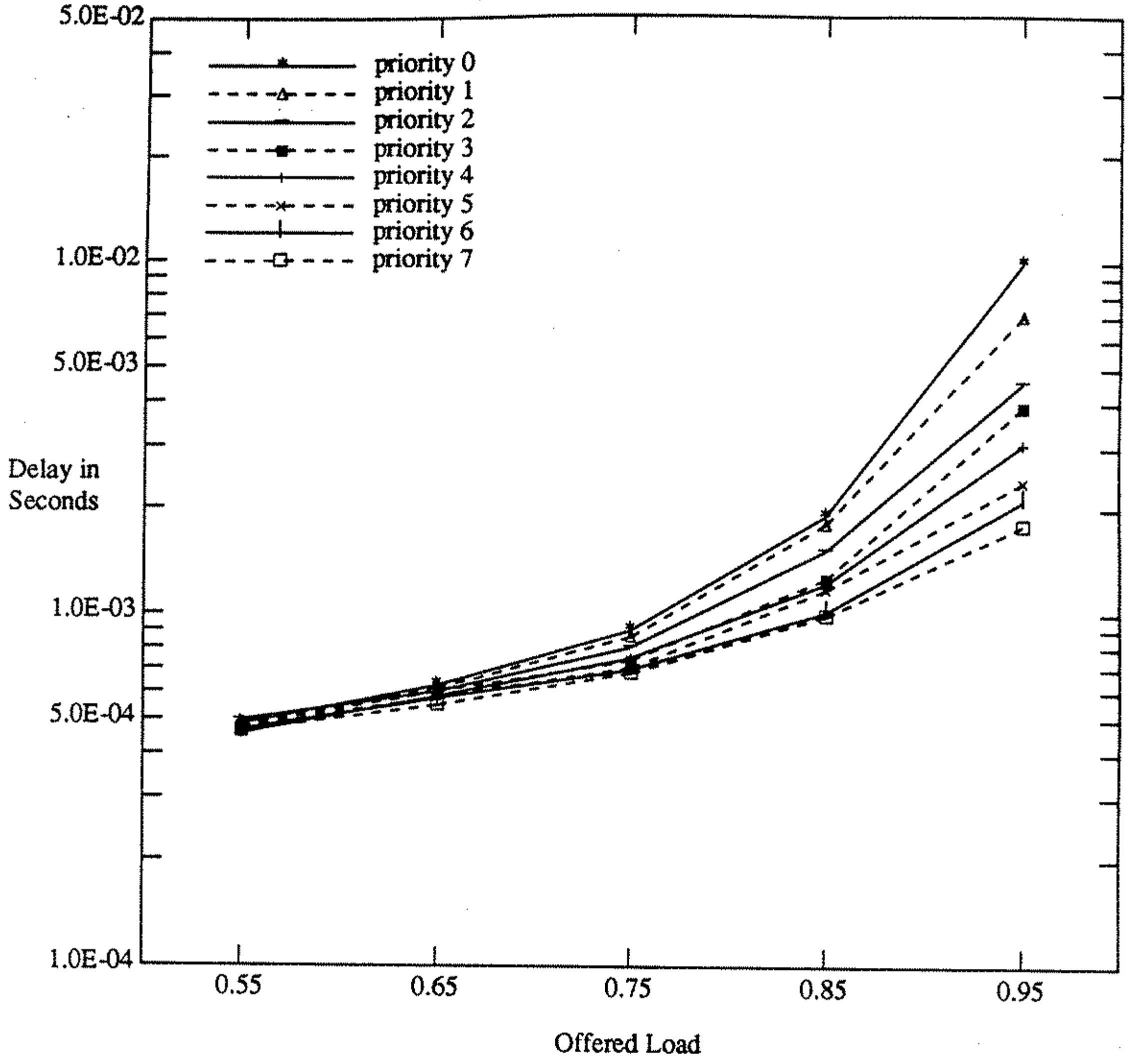

Figure 5
Delay for the SPPT service mode, 20 stations.

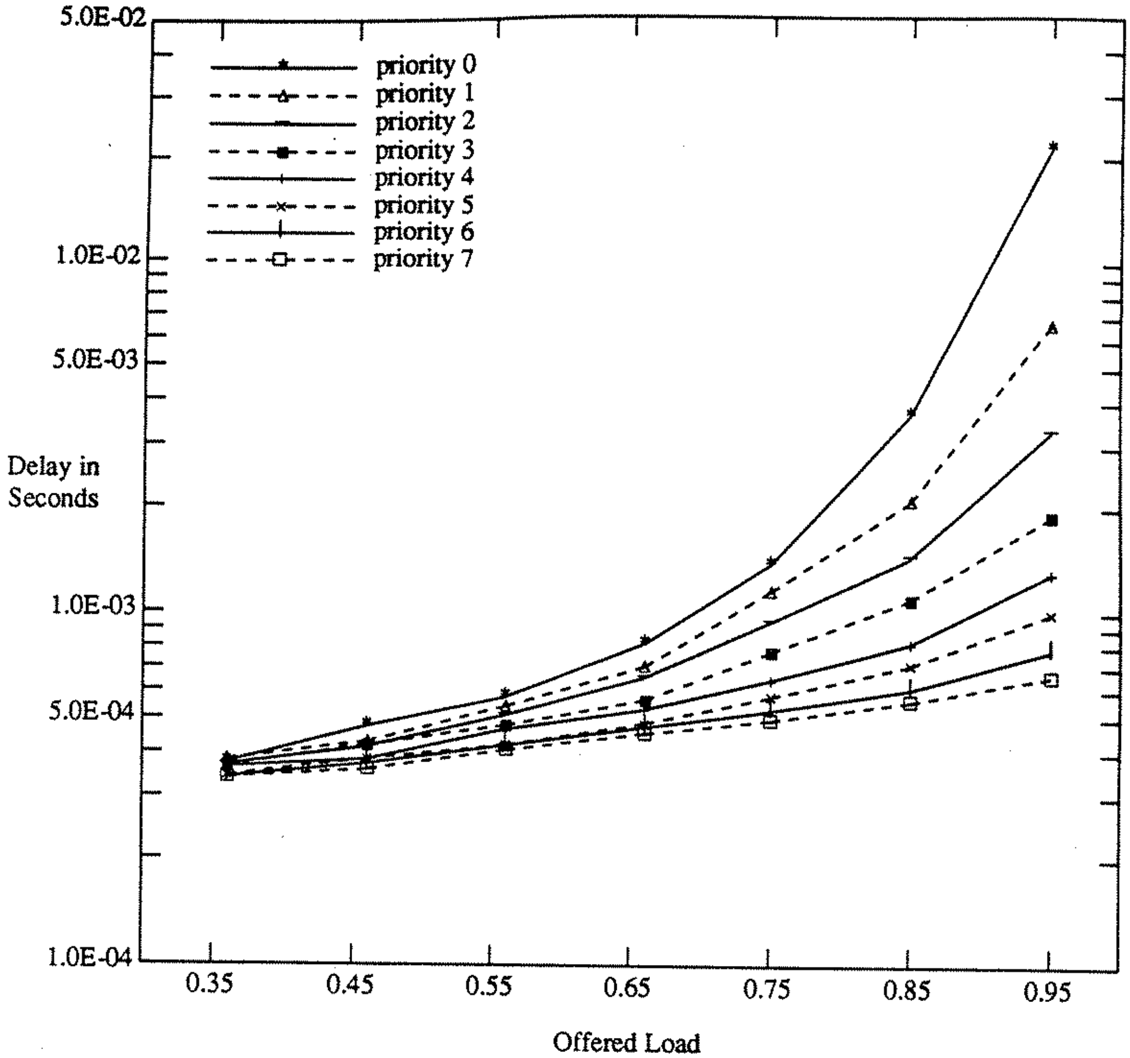

Figure 6
Delay for the SPPT service mode, 3 stations.

Offered Load	Delay by Priority 0	1	2	3	4	5	6	7
0.05	0.320 ms	0.320 ms	0.318 ms	0.320 ms	0.318 ms	0.319 ms	0.321 ms	0.319 ms
0.15	0.342 ms	0.341 ms	0.339 ms	0.339 ms	0.340 ms	0.339 ms	0.339 ms	0.337 ms
0.25	0.364 ms	0.362 ms	0.364 ms	0.363 ms	0.365 ms	0.364 ms	0.365 ms	0.365 ms
0.35	0.403 ms	0.398 ms	0.398 ms	0.397 ms	0.396 ms	0.399 ms	0.396 ms	0.400 ms
0.45	0.461 ms	0.460 ms	0.464 ms	0.461 ms	0.457 ms	0.458 ms	0.465 ms	0.462 ms
0.55	0.581 ms	0.571 ms	0.567 ms	0.560 ms	0.570 ms	0.571 ms	0.587 ms	0.564 ms
0.65	0.825 ms	0.831 ms	0.829 ms	0.822 ms	0.823 ms	0.831 ms	0.820 ms	0.816 ms
0.75	1.17 ms	1.15 ms	1.17 ms	1.16 ms	1.16 ms	1.16 ms	1.16 ms	1.13 ms
0.85	1.80 ms	1.83 ms	1.77 ms	1.80 ms	1.79 ms	1.76 ms	1.78 ms	1.74 ms
0.95	3.76 ms	3.76 ms	3.81 ms	3.80 ms	3.77 ms	3.72 ms	3.64 ms	3.63 ms

Table 1
Delays for multiple priority operation.

Offered Load	Delay	Offered Load	Delay
0.05	0.318 ms	0.55	0.505 ms
0.15	0.339 ms	0.65	0.609 ms
0.25	0.363 ms	0.75	0.779 ms
0.35	0.396 ms	0.85	1.20 ms
0.45	0.442 ms	0.95	3.18 ms

Table 2
Delays for single priority operation.