On The Behavior of Hubnet with Applications to Priority Implementation

Hossam S. Hassanein and *Ahmed E. Kamal*

Department of Computing Science,
University of Alberta,
Edmonton, Alberta T6G 2H1, Canada

Abstract

We study and analyze the behavior of Hubnet. A simulation model is developed from which the effect of the retry time on the performance of Hubnet is studied. Results show that increasing the retry time does not always mean higher delay. We show that, for retry time values that are less than and are factors of the packet length, the mean delay is better than some lower retry time values subject to some range. Also, we show that, contrary to what was believed earlier [1-4], a packet in a Hubnet with retry time which is a factor of the packet length is guaranteed transmission rights. We apply these results in studying a Hubnet with two priority classes, where the retry times are used for priority assignment. It is shown that a lower retry time does not always mean higher priority. Indeed, we show that a class of users with a retry time value that is a factor of the packet length could have higher priority than a class of users with a lower non-factor retry time value.

1. Introduction

Hubnet [1] is a collision-avoidance star local area network that was developed at the University of Toronto in Canada, in cooperation with CANSTAR corporation. Hubnet uses glass fiber as its transmission medium and operates at a data rate of 50 Mbits/s, that was recently upgraded to 100 Mbits/s.

Collision-Avoidance Star Local Area Networks (CASLANS) were discovered independently by Lee and Boulton [1], Closs and Lee [2], and Albanese [3]. The access protocol for CASLANS is based on repeated attempts by nodes to acquire the hub (central node). The protocol consists of two components, one executed at the nodes, the other at the hub.

Nodes are either idle (have no packets to transmit) or ready (have a packet that have not yet acquired the hub). The node protocol is as follows. As soon as an idle node receives a packet, it forwards that packet to the hub. A packet transmitted by a node acquires the hub if the hub is free upon the packet arrival at the hub. Otherwise, the packet is blocked. If a node does not sense its own packet after the round trip propagation delay between the node and the hub, it retries transmission immediately until successful (the round trip propagation delay between a node and the hub is thus called its retry time).

The hub protocol is as follows. If a packet arrives at the hub and the hub is free, the packet acquires the hub instantaneously. Once a Packet acquires the hub, the packet is broadcast on all outgoing links. If two or more packets arrive at the hub at the same time, one of them is chosen at random and the other packets are blocked. Thus, collisions never occur in such networks, and consequently avoiding excessive bandwidth wastage and guaranteeing a stable behavior.

It should be noted that, even though this protocol is collision free, and hence achieves very high utilization, it does not prevent unbounded delays. Indeed, depending on the offered load and the system parameters, it is possible for a particular packet to always encounter a busy hub, thus resulting in an infinite delay.

Several attempts were made to study the performance of Hubnet. The attempts varied from analytical performance models [4-9] to actual performance measurements [10]. None of these attempts was directed to studying the effect of the propagation delay (retry time) on the network performance. In this paper we study and analyze the effect of node retry times on the average delay and throughput of Hubnet.

It is through this study that we discovered an inherent interdependence between the packet length and the retry time. The results in this paper show that, contrary to what was believed earlier, increasing the retry time does not always mean an increase in delay. In fact, we show that increasing the retry time to a value which is a factor of the packet transmission time could result in a much better performance.

The remainder of this paper is arranged as follows. Section 2 contains results obtained using a simulation model, and show the effect of the retry time on the average delay of Hubnet. These results were a partial motivation behind the work in this paper.

In section 3, we analyze the behavior of Hubnet in which the retry time is a factor of the packet length. We show that not only does these retry times result in a better performance, but also a packet in such a system is guaranteed transmission rights.

In view of the results in sections 2 and 3, in section 4, we present a priority study of a two-class Hubnet in which the retry times are used for priority assignment. Finally, in section 5, we provide some concluding remarks.

2. Simulation Results

In this section, we show and analyze simulation results of Hubnet where all nodes are identical. That is, all nodes have the same retry time and mean packet length.

Our simulation model behaves exactly like the Hubnet protocol under the following conditions:

1) All transmitting nodes are equipped with single buffers. That is, no new packets are generated at a node unless the

This research was supported in part by NSERC (Canada) Grant OGP-9187.

packet in the buffer has been successfully transmitted.

2) No packet loss due to network failure or buffer overflow at the receiving end is considered.

3) The transmission medium is noiseless and error free.

4) Two types of packets are considered: variable length and fixed length packets.

5) The round trip propagation delay between any node and the hub is the same for all nodes, and is equal to the retry time.

6) The delay of a packet is measured from the time the packet is generated until the packet arrival at the destination.

Figure 1 shows the average delay versus the retry time of a symmetric Hubnet with N = 20 nodes and a variable packet length, at different values of the packet arrival rate per node, λ. As expected, as the retry time increases the delay increases.

Figure 2 shows the average delay versus the retry time for a symmetric Hubnet with N=20, with a fixed packet length (T = 480) at different values of λ. At light load (e.g. $\lambda = 10^{-5}$, offered load = 0.096) the greater the retry time, the greater the delay. This result does not always hold at higher loads. For example, at $\lambda = 0.001$ (offered load = 9.6), the average delay at $\tau = 100$ is 179 μs, whereas at $\tau = 120$ the average delay is 173μs. That is, the delay at a higher value of retry time (120) is less than the one at a lower retry time (100). This result, which defies intuition at first glance, is due to the fact that 480 (packet length) is a multiple of 120 (retry time). The same phenomenon is observed for every τ which is a factor of T (such as 160, 240 and 480). At all of these τ values, a drop in the mean packet delay is observed, which results in a local minimum of the average delay at these values. Similar results were observed for other values of N.

In the remainder of this paper, we treat the case where the retry time is less than or equal to the packet length. To show why such a case is a practical one, consider the following example. For Hubnet, where the minimum packet length is 88 bits, and with a bit rate of 100 Mps, that results in a network with a radius of about 130 meters (for local networks distributed within a building or some adjacent buildings, this distance is very reasonable). We also confine our study to the case where the packet length is fixed. This is due to the following

1) Many applications such as voice communication and uncompressed video communication require fixed packet length transmission.

2) For the variable packet length case, the mean packet delay increases with the retry time. Therefore, studying the effect of the retry time on the performance in such a case does not provides any new insight into the behavior of the network.

We begin our study of the effect of the retry time on the performance of Hubnet, by making the following observation from our simulation results.

Proposition I:
For the fixed packet length (T) Hubnet, for every τ_f that is a factor of T, there exists a period $[\tau_f - \delta, \infty)$ in which $D(\tau_f) \leq D(\tau)$, where $\tau \in [\tau_f - \delta, \infty)$ and D(X) denotes the average delay at retry time X. We call the period $[\tau_f - \delta, \infty)$ the *range* of τ_f.

□

That is to say, if the retry time is a factor of T, then the mean packet delay at that retry time is better than every higher retry time and some lower retry times subject to its range. This proposition is based on observations from simulation results. We do not provide a proof for it nor do we provide a general expression for the range of a factor retry time value.

The width of that range, and in particular δ, is dependent on the load and the number of nodes. For instance, at very light load $\delta = 0$ (see figure 2, at $\lambda = 10^{-5}$), since the delay always increases with τ. At heavy load, however, $\delta > 0$. For instance, at $\lambda = 0.001$ (offered load = 9.6), the range of 240 is approximately $[170, \infty)$.

A direct consequence of proposition I is that if τ_1 and τ_2 are two factors of T and $\tau_1 < \tau_2$, then $d(\tau_1) \leq d(\tau_2)$. This follows from the non-existence of an upper bound on the range of τ_1. This fact is confirmed by the results in figure 2, $D(120) \leq D(160) \leq D(240)$, and is proven in the following section.

In view of the above results, it is appropriate that we study the following modes of operation of Hubnet.

1) Hubnet with retry time that is a factor of the packet length. The objective is to find why such retry times are special.

2) Since rertry time values that are are factors of the packet length seem to provide superior performance, their use in priority implementation in Hubnet is considered.

3. Hubnet with a Retry Time That is a Factor of Packet Length

In this section, we study and analyze the behavior of Hubnet in which the retry time is a factor of the packet length. Let τ (the retry time) be a factor of T (the packet length) such that T is constant, and $T = k \cdot \tau$, where $k > 0$ is an integer number.

We begin by making the following definitions. Recall that the Hubnet protocol is based on repeated attempts by the nodes to acquire the hub. If a retry attempt arrives at the hub during the same hub acquisition as the previous one (i.e. the hub has not yet been released), we call this arrival an *intrapacket* arrival. If the retry attempt arrives at the hub during a following hub acquisition, we call it an *interpacket* arrival. Therefore, for any node, and during any hub acquisition period, there is exactly one interpacket arrival and several intrapacket arrivals. We denote the remainder of the hub acquisition time on the l^{th} interpacket arrival of a certain packet by X_l, and the time from the start of the hub acquisition by Z_l. We call the latter the l^{th} interpacket arrival time. Obviously, $Z_l = T - X_l$. We denote the remainder of the hub acquisition time at the point of the last intrapacket arrival by Y_l. The overhead period between hub acquisitions is denoted by t, see figure 3. It should be noted that the values of X_1 and Z_1 are not very meanningful. This is because a packet may arrive randomly during the hub acquisition on its first interpacket arrival, see X_1 in figure 4.

Theorem 1:
If X_{l-1}, X_l, Z_{l-1} and Z_l are as defined above and τ is a factor of T, then for every packet being transmitted

$$1)\ X_l \geq X_{l-1}, \text{ and } 2)\ Z_l \leq Z_{l-1}$$

Proof:
1) From figure 3, with $T = k \cdot \tau$, we see that

$$X_l = T + t + Y_{l-1} - \tau$$

But

$$Y_{l-1} = X_{1-1} - (k-1)\cdot\tau$$

Thus

$$X_l = T + t + X_{l-1} - (k-1)\cdot\tau - \tau$$
$$= T + t + X_{l-1} - k\tau$$
$$= t + X_{l-1}$$

Since t, the overhead between transmissions is ≥ 0, then

$$X_l \geq X_{l-1}$$

2) Since $Z_l = T - X_l$, then it follows from part (1) that $Z_l \leq Z_{l-1}$.

□

The above theorem implies that the interpacket arrival time of a transmitted packet keeps getting smaller[1] with the number of interpacket retry attempts. An interpacket arrival time of $Z_l = 0$ means that the packet has acquired the hub.

Using theorem 1 above, we next prove that if τ_1 and τ_2 are both factors of T, and if $\tau_1 < \tau_2$, then the average value of the second interpacket arrival time is less in the case of $\tau = \tau_1$. We first introduce $X_l(\tau)$, $Y_l(\tau)$ and $Z_l(\tau)$ to denote the value of X_l, Y_l and Z_l in a Hubnet with retry time τ.

Theorem 2:
Let τ_1 and τ_2 be both factors of T. If $\tau_1 < \tau_2$, then

$$E[Z_2(\tau_1)] \leq E[Z_2(\tau_2)]$$

Proof:

We make the following assumptions:

1) The packet generation process is Poisson.
2) The length of the overhead period (t) has the same distribution whether the retry time is τ_1 or τ_2. We do not make any assumption about this distribution, and we leave it to be a general one.

Let W denote the time of arrival of a tagged packet on its second interpacket arrival, measured from the last hub release. Thus, $W = Z_2 + t$, see figure 4. Let $f_W(u, \tau)$ denote the pdf of W in a system with retry time τ. $f_W(u, \tau)$ is given by (1) below.

$$f_W(u,\tau) = \sum_{i=1}^{k} f_{X_1}(i\tau - u) \tag{1}$$

Where, $f_{X_1}(v)$ is the pdf of the remainder of the hub acquisition time upon a packet arrival from an idle node and is given by

$$f_{X_1}(v) = \frac{\lambda e^{-\lambda(T-v)}}{1-e^{-\lambda T}} \tag{2}$$

Therefore,

$$f_W(u,\tau) = \sum_{i=1}^{k} \frac{\lambda e^{-\lambda(T-i\tau+u)}}{1-e^{-\lambda T}}$$

$$= \frac{\lambda e^{-\lambda u}}{1-e^{-\lambda\tau}} \tag{3}$$

Now, let $W(\tau)$ and $t(\tau)$ respectively denote the values of W and t in a system with retry time τ.

$$E[W(\tau)] = \int_0^\tau u\cdot f_W(u,\tau)\cdot du$$
$$= \int_0^\tau u\cdot\frac{\lambda e^{\lambda u}}{1-e^{\lambda\tau}}\cdot du$$
$$= \frac{1}{\lambda} - \frac{\tau}{e^{\lambda\tau}-1} \tag{4}$$

Now if $\tau_1 < \tau_2$, then it is not hard to see that

$$\frac{\tau_1}{e^{\lambda\tau_1}-1} \geq \frac{\tau_2}{e^{\lambda\tau_2}-1} \tag{5}$$

From (4) and (5)

$$E[W(\tau_1)] \leq \frac{1}{\lambda} - \frac{\tau_2}{e^{\lambda\tau_2}-1} = E[W(\tau_2)] \tag{6}$$

Since

$$W(\tau) = t(\tau) + Z_2(\tau)$$

Then

$$E[W(\tau)] = E[t(\tau)] + E[Z_2(\tau)] \tag{7}$$

Substituting in (6) above, we get

$$E[t(\tau_1)] + E[Z_2(\tau_1)] \leq E[t(\tau_2)] + E[Z_2(\tau_2)]$$

Since, according to assumption 2, the overhead period, t, has the same distribution with retry times τ_1 and τ_2, then $E[t(\tau_1)] = E[t(\tau_2)]$. Thus

$$E[Z_2(\tau_1)] \leq E[Z_2(\tau_2)] \tag{8}$$

□

Let D_Z denote the remainder of the packet delay at the hub starting from the second interpacket arrival of a tagged packet (see figure 4). In the following, we show that $E[D_Z \mid Z_2]$ is proportional to the value of Z_2.

Theorem 3:
$E[D_Z \mid Z_2]$ increases with the value of Z_2.

Proof:

Let n be the number of interpacket transmissions before a user acquires the channel. That is, the packet acquires the channel on its $(n+1)^{st}$ interpacket arrival, i.e. $Z_{n+1} = 0$.

$$E[D_Z \mid Z_2] = \sum_{n=2}^{\infty} \int_{t_n}\ldots\int_{t_2} (nT - Z_2 + u_2 + u_3 + \ldots + u_n)$$
$$\times f_{t_2,t_3,\ldots,t_n}(u_2,u_3,\ldots u_n;n)\cdot d_{u_2}\ldots d_{u_n} \tag{9}$$

Recall from theorem 1 that $Z_{l-1} = Z_l + t_{l-1}$. That is,

$$Z_2 = Z_3 + t_2$$
$$= Z_4 + t_2 + t_3$$

1 An exception would be the case where the hub is never idle. In this case the interpacket arrival does not decrease and will never reach 0 (successful transmission). Such a situation, however, is almost practically impossible since it implies that the processing time at the hub is always zero, and that packet transmissions are synchronized.

$$= Z_{n+1} + \sum_{i=2}^{n} t_i = \sum_{i=2}^{n} t_i \quad (10)$$

Substituting in (9), we have

$$E[D_Z | Z_2] = \sum_{n=2}^{\infty} \int_{t_n} \ldots \int_{t_2} (nT - Z_2 + Z_2) \times f_{t_2,t_3,\ldots,t_n}(u_2, u_3, \ldots u_n; n) \cdot d_{u_2} \ldots d_{u_n}$$
$$= T \sum_{n=2}^{\infty} n \int_{t_n} \ldots \int_{t_2} f_{t_2,t_3,\ldots,t_n}(u_2, u_3, \ldots u_n; n) \cdot d_{u_2} \ldots d_{u_n}$$
$$= T \cdot E[n | Z_2] \quad (11)$$

From equation (10) above, it is not hard to see that $E[n | Z_2]$ increases with Z_2. Hence $E[D_Z | Z_2]$ increases with Z_2.

□

An interesting observation here is that the above result is independent of τ[2]. That is, if τ_1 and τ_2 are both factors of T and $Z_2(\tau_1) \le Z_2(\tau_2)$, then $D_Z(\tau_1) \le D_Z(\tau_2)$.

Lemma 1:
Let τ_1 and τ_2 be both factors of T. If $\tau_1 < \tau_2$, then

$$E[D_Z(\tau_1)] \le E[D_Z(\tau_2)]$$

Proof:

We begin the proof by augmenting assumption 2 (in theorem 2 above) by adding that the overhead periods are statistically independent of each other. Let $\bar{t}$ be the average length of an overhead period. Thus

$$E[n | Z_2] = \frac{Z_2}{\bar{t}}$$

that is

$$E[n] = \frac{E[Z_2]}{\bar{t}}$$

From (11) and by unconditioning on Z_2, we get

$$E[D_Z] = T \cdot E[n] = T \cdot \frac{E[Z_2]}{\bar{t}}$$

Thus

$$E[D_Z(\tau)] = T \cdot \frac{E[Z_2(\tau)]}{\bar{t}}$$

From theorem 2, we know that if $\tau_1 < \tau_2$ and τ_1 and τ_2 are both factors of T, then

$$E[Z_2(\tau_1)] \le E[Z_2(\tau_2)]$$

Therefore,

$$E[D_Z(\tau_1)] \le E[D_Z(\tau_2)]$$

□

From Figure 4, we can see that the average delay ($\bar{D}$) is given by (12) below

$$\bar{D}(\tau) = E[X_1(\tau)] + E[W(\tau)] + E[D_Z(\tau)] + T + \tau \quad (12)$$

If we assume that $E[X_1(\tau)]$ is independent of τ, then from theorem 2 and lemma 1 above, for τ_1 and τ_2 both factors of T, and $\tau_1 < \tau_2$, $\bar{D}(\tau_1) < \bar{D}(\tau_2)$. That is, theorems 2 and 3 along with lemma 1 above seem to suggest that the range of a factor retry time value cannot extend to include the greatest lower factor retry time. For instance, referring to Figure 2, $\bar{D}(120) < \bar{D}(160) < \bar{D}(240) < \bar{D}(480)$, where $\bar{D}(\tau)$ is the average delay at retry time (τ).

The following theorem proves that if τ is a factor of T, then every packet transmitted will eventually be successful. That is, no packet could face a busy hub indefinitely.

Theorem 4:
Let τ be a factor of T and let Q be a packet transmitted over the channel using τ. Q will acquire the hub with probability 1.

Proof[3]*:*

Let ε be the smallest time unit distinguishable by the hub, such that ε is very close to zero. Note that the range of X_l is $(T - \tau, T]$. Let the range of X_l be divided into m mini-periods each of length ε. We can see that the remainder of a packet transmission time is a Markovian chain {n, n = 0, 1, ..., m} on a state space consisting of integer numbers between 0 and m. Let $P_{i,j}$ represent the transition probability from state i to state j (this means that $X_l - X_{l-1} = (j - i) \cdot \varepsilon$).

From theorem 1, we see that $P_{i,j} = 0$ if $j < i$. Thus

$$\sum_{j=i}^{m} P_{i,j} = 1$$

We next show that state m is an absorbing state and that the probability that leaving any state i, $i < m$, we will eventually reach state m is equal to 1. We prove this by mathematical induction.

The first passage probability to state m given that the system was in state i, $f_{i,m}$, is given by

$$f_{i,m} = P_{i,m} + \sum_{j=i}^{m-1} P_{i,j} \cdot f_{j,m} \quad (13)$$

Now

$$f_{m-1,m} = P_{m-1,m} + P_{m-1,m-1} \cdot f_{m-1,m}$$
$$= \frac{P_{m-1,m}}{1 - P_{m-1,m-1}} = 1 \quad (14)$$

Let us assume that $f_{j,m} = 1$, for $j \ge k$. Then

$$f_{k-1,m} = P_{k-1,m} + \sum_{j=k-1}^{m-1} P_{k-1,j} \cdot f_{j,m}$$
$$= \frac{P_{k-1,m} + \sum_{j=k}^{m-1} P_{k-1,j} \cdot f_{j,m}}{1 - P_{k-1,k-1}}$$
$$= \frac{\sum_{j=k}^{m} P_{k-1,j}}{1 - P_{k-1,k-1}} = 1 \quad (15)$$

2 This is true only if assumption 2 holds. In practice, t is a function of τ. However, we assume that all users behave in a way, and that the tagged user behaves in a distinct way.

3 We prove this theorem using a discrete, yet infinitismal approach. The contineous case, however, can be treated by considering the limit as $m \to \infty$.

From (14) and (15) above, the theorem follows.

□

Thus $f_{i,m} = 1$, for $i = 0, 1, ..., m-1$. Since state m represents a successful transmission, this proves that, regardless of the point of arrival of a packet at the hub, it is guaranteed successful transmission if τ is a factor of T.

Before concluding this section, we make the following comments

1) A retry time value that is a factor of the packet length was shown to provide some superior performance results. Indeed, not only does a retry time that is a factor of the packet length provide packets with always decreasing inter-packet arrival times, but it also guarantees these packets an eventual transmission. Such advantages were reflected in the simulation results provided earlier, see section 2.

2) The relations and facts proved in this section apply regardless of the retry times with which other packets in the system are transmitted by. In fact, we show in the following section that improved performance, of retry time values that are factors of the packet length, holds for the two-class and subsequently for multiple-class Hubnet.

4. Hubnet with Two Priority Classes

In this section we consider the Hubnet with two classes, each of which has its own mean packet delay, arrival rate and retry time. We analyze the effect of the retry time on the average delay of each class and show use of retry times in priority assignment. In doing this, we make use of some of results obtained in sections 2 and 3.

Let C_1 and C_2 denote classes 1 and 2 respectively. Let N_i, T_i, λ_i and τ_i respectively be the number of nodes, packet length, arrival rate and the retry time for C_i, where $i = 1$ or 2. We start by studying the fixed packet length case, $T_1=T_2=T$. For simplicity, we let $\lambda_1=\lambda_2=\lambda$.

Figures 5 and 6 show the average delay versus τ_2 for $N_1=N_2=10$, and T = 480, for various values of λ at $\tau_1= 110$ (a non-factor of T) and 120 (a factor of T) respectively. By analyzing these results, as well as results at different values of T and N, we make the following observations

1) At light load, the higher the retry time, the higher the average delay. Subsequently, the class with the lower retry time has higher priority.

2) At high load, increasing the retry time for either class does not always mean higher average delay. Indeed, at retry time values that are factors of the packet length, a decrease in average delay in the form of local minima is observed. This conforms to the single class results (section 2).

3) The observed decrease in average delay of a class of users is associated with an increase in average delay of the other class, even though its retry time has not changed (we comment on this later).

From the above results, we make the following proposition.
Proposition II:
Let τ_1 be a factor of T, and let $D_1(\tau)(D_2(\tau))$ denote the delay of $C_1(C_2)$ at τ. Then, there exists a period $[\tau_1-\delta,\infty)$ in which $D_1(\tau_1)\leq D_2(\tau_2)$, where $\tau_2 \in [\tau_1-\delta,\infty)$.

□

The value of δ increases with the load and reduces to 0 at very light load. At very light load $D_1(\tau_1)\leq D_2(\tau_2)$ if and only if $\tau_1\leq\tau_2$. On the other hand, the range of τ_f (a factor of T) could, at heavy load, extend to the greatest factor of T less than τ_f. Note that the above proposition suggests that if τ_1 is a factor of T, then there exists no $\tau_2>\tau_1$ such that C_2 would have a higher priority over C_1. This proposition is equievelant to proposition I for the single class case. Proposition II, however, is concerned with comparing the retry times of two contending classes of users. On the other hand, proposition I compares two modes of system operation, each under a different retry time value. The following results follow from proposition II.

Corollary 1:
If τ_2 is a factor of T, then C_1 could only have higher priority than C_2 if $\tau_1<\tau_2$.
Proof:
Since τ_2 is a factor of T, then its range includes all values higher than itself. Thus, for C_1 to have higher priority over C_2, τ_1 must be less than τ_2.

□

Corollary 2:
If τ_2 is a factor of T, then (in the absence of information about δ) the largest τ_1 value to be assigned to C_1 and still guarantee it higher priority over C_2 is equal to the greatest factor of T less than τ_2.
proof:
Let τ be the assigned value for τ_1. If τ is a factor of T then its range would include every value greater than itself. Let τ_g be the greatest factor of T less than τ_2. If $\tau = \tau_g$, then τ_2 is in the range of τ and C_1 would have higher priority. Now, let τ, $\tau_g<\tau<\tau_2$ be a non-factor of T. Since the range of τ_2, $[\tau_2-\delta,\infty)$, may include τ, then assigning $\tau_1=\tau$ does not guarantee a higher priority for C_1.

□

In order to empirically verify the above corollaries, consider figures 5 and 6. In figure 5 with $\tau_1=110$, $\lambda=0.0001$ and $\tau_2=160$, $D_1(\tau_2)< D_2(\tau_1)$. In figure 6, $\tau_1=120$ and at the same values of λ and τ_2, we find that $D_1(\tau_1) < D_1(\tau_2)$. That is, by increasing τ_1 from 110 to 120, C_1 is assigned a priority that is higher than C_2. This result also shows that for τ_1, being less than τ_2, does not guarantee a higher priority for C_1. Only be making τ_1 a factor of T ($\tau_1= 120$), priority over C_2 was guaranteed. Also note that with $\tau_1 = 120$, C_1 always has lower average delay than C_2 for τ_2 in the range [100, 480], see figure 6. In fact, the range of $\tau_1 = 120$, at offered load = 9.6, extends to the greatest lower factor retry time (96) and is approximately given by $[96.3, \infty)$.

It should be noted that the effect of changing the retry times is much more apparent here than in the single class case (section 2). For instance, in figure 5, changing τ_2 from 240 to 180 at $\lambda=0.001$ yields an increase of 105% in average delay, as appossed to an increase of only 10% in the single class case. The reason is that as τ_2 changes such that C_2 has lower priority, C_1 packets will face less contention at the hub causing a decrease in $D_1(\tau_1)$ and C_2 packets will face more of C_1 packets, this causes more C_2 packet retries. This results in more C_1

packets being transmitted, and so on. We call this phenomenon where one class monopolizes the hub as *hub hugging*. To illustrate the effect of hub hugging, we compare the number of successfully transmitted packets of C_1 and C_2 at different values of τ_1 and τ_2, see Table 1. In Table 1, λ=0.001, T=480 and a total of 50,000 packets were successfully transmitted over the simulation time. From the table we can see that hub hugging is strongest at $\tau_1(\tau_2)$ values which are factors of T. For instance, at τ_1=120 and τ_2=280, 40,559 C_1 packets and only 9441 C_2 packets are transmitted. With $\tau_1 = 110$ and $\tau_2 = 280$, these numbers changed dramatically to 32,130 packets from C_1 and 17,870 packets from C_2.

An implication of the apparent increase of hub hugging at retry time values that are factors of the packet length, is that a priority class using such a retry time is not severely affected by the presence of packets from another class that uses a non-factor retry time value. By examining $D_1(120)$ at offered load = 9.6, $N_1 = N_2 =$ 10 and with τ_2 in the range [280, 440]_ see figure 6_ we can see that the average delay is not much higher than that at $N_2 = 0$ (Hubnet with N = 10) where D(120) = 81.5μs.

We have shown that assigning a factor retry time value to a priority class results in lower average packet delay. In some cases such as real-time applications, however, reasonable average performance does not suffice. In real-time application, the probability of packets being received before their deadlines is the primary performance measure. This makes controlling the maximum delay as important as, and may be more important than, maintaining a low average delay. In figure 7, we show the maximum delay of C_1 (real-time traffic) versus τ_1 at various values of τ_2 (where C_2 is a non-real-time traffic) with $N_1 = N_2 = 10$, T = 480 and an offered load of 9.6.

From the results in figure 7, it is not hard to notice the much lower maximum delay at retry time values that are factors of the packet length (120, 160, and 240). For instance, at $\tau_2 = 140$, the maximum delay of C_1 decreased from 2460 μs to 560 μs when τ_1 was increased from 100 to 120. Also note that a low maximum delay is maintained at retry time values that are factors of the packet length, even with τ_2 a factor of the packet length, see the delay at $\tau_2 = 160$ in figure 7.

Finally, we study the case where $T_2 \neq T_1$. We choose T_2 to be exponentially distributed, which eliminates the effect of T_2 on the priority assignment. In figure 8 we show the average delays of C_1 and C_2 versus the retry time of C_1 at τ_2=110 N_1=N_2=10, and $T_1 = T_2 = 480$. From analysis of the results of figure 8, we can make the following observations.

Proposition II and corollaries 1 and 2 still hold in the case where $T_1 \neq T_2$. However, the effect of the factor retry times is not so profound. A decrease of only 21.5% is noted in D_1 when decreasing τ_1 from 140 to 120 at λ=0.001 and τ_2=110, whereas in the fixed packet length case the decrease was 39%. This is due to the fact that $T_2 \neq T_1$ and that T_2 is variable causing the interdependence between the packet length and retry time to decrease. However, this interdependence did not vanish because T_1 is still fixed and τ_1 is a factor of T_1. Therefore, this still gives C_1 higher priority over C_2 within the range of τ_1.

The results in figures 7 and 8 are of extreme importance, since they show the potential of Hubnet towards integrated services (e.g. voice and data). Since voice packets are of fixed length, they could be transmitted with a retry time that is a factor of the packet length. Variable length data packets could be transmitted with a retry time, a non-factor of the packet length that is greater than the retry time for voice packets. This would not only assign higher priority to voice packets, but also results in a very low percentage of lost voice packets. This is because of the much lower maximum delay associated with retry time values that are factors of the packet length.

5. Conclusion

In this paper we conducted an in-depth study on the effect of the retry time on the behavior of Hubnet. Also, a two-class Hubnet was analyzed through a simulation model.

Simulation results show that, in contrast to what was believed earlier, increasing the retry time does not always mean an increase in delay. In fact, the results show that increasing the retry time to a value which is a factor of the packet transmission time could result in a much better performance.

By analyzing the behavior of Hubnet in which the retry time is a factor of the packet length, the following has been shown.

1) The average and maximum delays at a retry time that is a factor of the packet length, is better than at every higher retry time, and some lower retry times subject to its range.
2) A packet in a Hubnet with retry time that is a factor of the packet length, is guaranteed to be transmitted.

Results of the two-class Hubnet priority study show that assigning a retry time that is a factor of the packet length provides higher priority over any greater retry time, and some lower retry times subject to its range. The results also show that Hubnet is a good candidate for integrated services.

References

[1] E. S. Lee and P. I. P. Boulton, "The Principles and Performance of HUBNET: A 50 Mb/s Fiber Optics Local Area Network," *IEEE Jour. Selec. Areas Comm.*, Vol. SAC-1, Nov. 1983, pp. 711-720.

[2] F. Closs and R. P. Lee, "A Multi-Star Broadcast Network for Local Area Communications Networks," *Local Networks for Computer Communications*, A. west and P. Jansen (eds.), North Holland, 1981, pp. 61-80.

[3] A. Albanese, "Star Networks with Collision Avoidance," *BSTJ*, Vol. 62, No.3, Mar. 1983, pp.631-638.

[4] T. Suda, Y. Yemini and M. Schwartz, "Tree Networks with Collision Avoidance Switches," *In Proc. of INFOCOM* 1984, pp. 105-113.

[5] V. C. Hamacher and W. M. Loucks, "Packet Transfer Delays in Local Area Computer Networks," *INFOR*, Vol. 23, No. 3, Aug. 1985, pp. 248-274.

[6] A. E. Kamal and V. C. Hamacher, "Analysis of a Star Local Area Network with Collision Avoidance," *In Proc. of INFOCOM* 1986, pp. 546-555.

[7] A. E. Kamal, "A Performance Model for a Star Network," *In Proc. of GLOBECOM* 1986, pp. 12-18.

[8] A. E. Kamal, "Star Local Area Networks: A Performance Study," *IEEE Trans. on Computers*, Vol. C-36, No. 4, April 1987, pp. 483-499.

[9] T. Suda and K. Goto, "Performance Study of a Tree LAN with Collision Avoidance," *In Proc. of INFOCOM* Apr. 1989, pp 59-68.

[10] E. S. Lee, P. I. P. Boulton, and B. W. Thomson, "HUB-NET Performance Measurements," *IEEE Jour. Selec. Areas Comm.*, Vol. SAC-6, No. 6, 1988, pp. 1025-1032.

τ_1/τ_2 (bits)	No. of packets of C_1	No. of packets of C_2
110/ 140	26,350	23,650
110/ 160	20,469	29,531
110/ 280	32,130	17,870
120/ 140	35,791	14,209
120/ 160	27,620	22,380
120/ 280	40,559	9,441

Table 1: The hub hugging effect

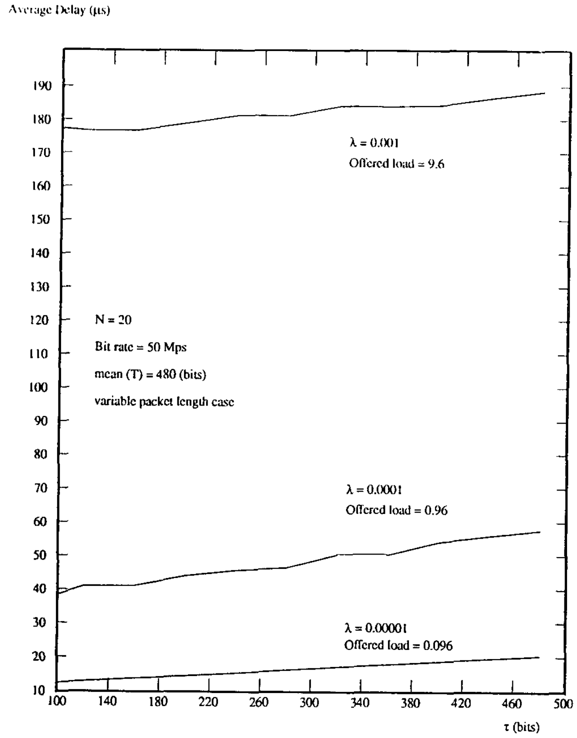

Figure 1: Response time at different values of τ in a 20 node system

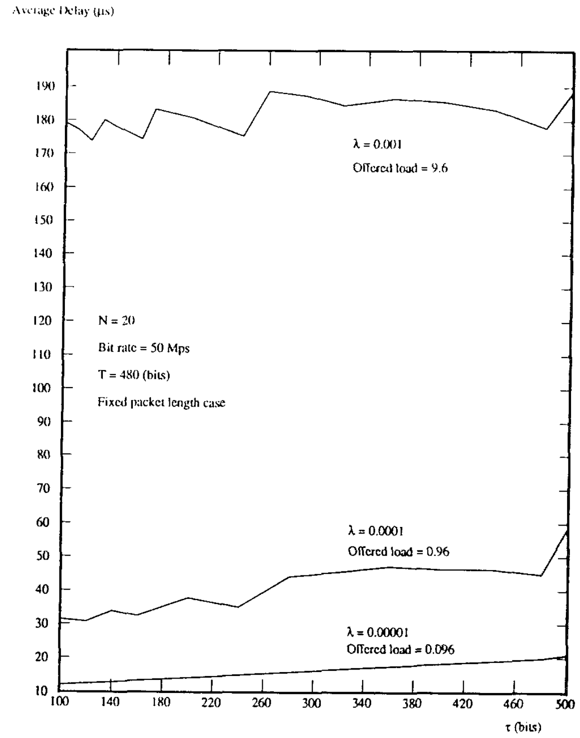

Figure 2: Response time at different values of τ in a 20 node system

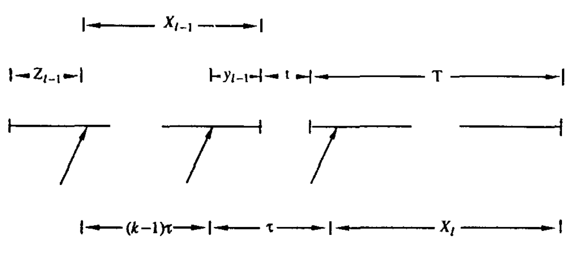

Figure 3: Relation between the $l-1^{st}$ and the l^{th} interpacket arrivals of a packet

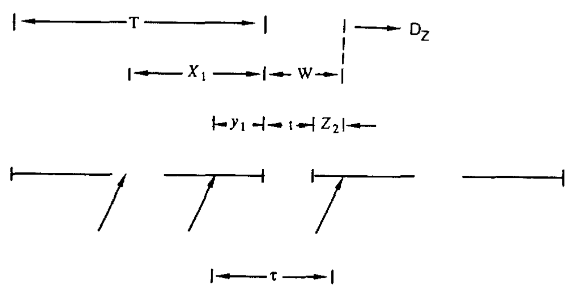

Figure 4: Relation between X_1, Z_2 and W

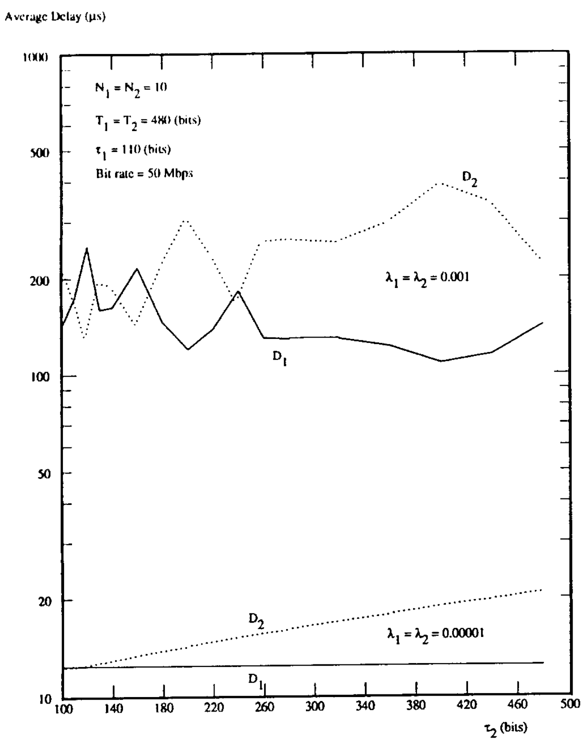

Figure 5: Response time at different values of τ_2 in a 20 node system.

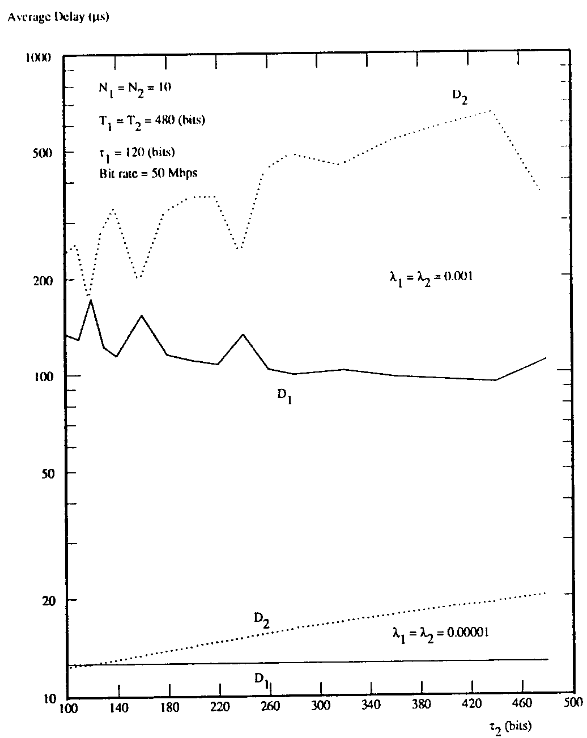

Figure 6: Response time at different values of τ_2 in a 20 node system.

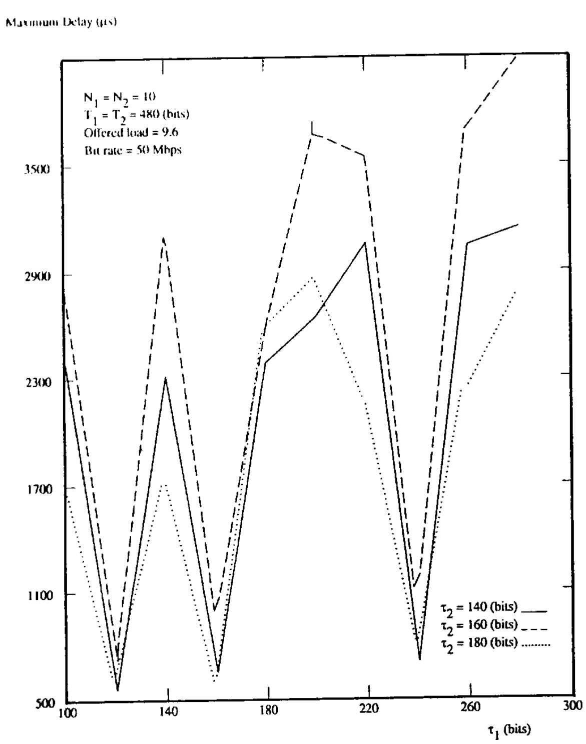

Figure 7: Maximum delay at different values of τ_1 in a 20 node system.

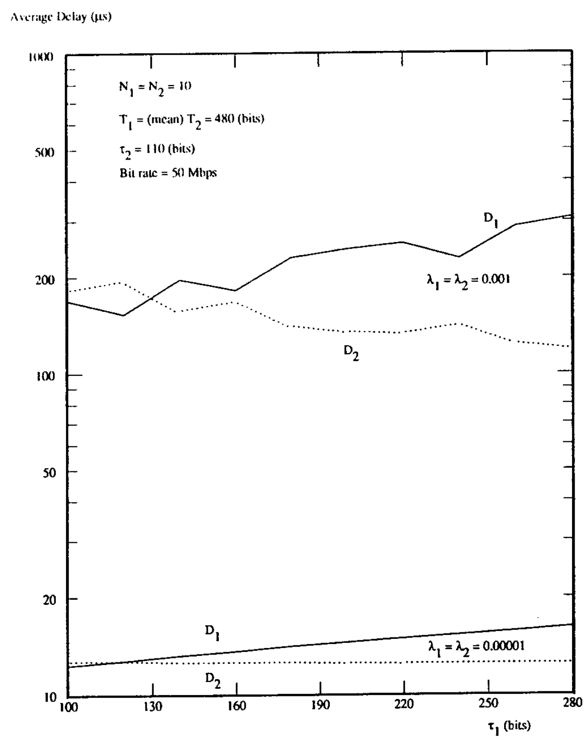

Figure 8: Response time at different values of τ_1 in a 20 node system.