# Reprint

## Performance Analysis of 100 Mbps PACE Technology Ethernet Networks

*A. Pantazi and T. Antonakopoulos*

The 6th IEEE Symposium on Computers and Communications-ISCC'01

# Performance Analysis of 100 Mbps PACE Technology Ethernet Networks

Aggeliki Pantazi[1] and Theodore Antonakopoulos[2]
[1] *Computers Technology Institute, Riga Feraiou 61, 26110 Patras, Greece*
[2] *Department of Electrical Engineering and Computers Technology,*
*University of Patras, 26500 Rio - Patras, Greece*
*Tel: +30-61-997 346, Fax: +30-61-997 342, e-mail: antonako@ee.upatras.gr*

## Abstract

*100BASE-T has enhanced the performance of Ethernet networks from 10 to 100 Mbps. However, an Ethernet network under high offered load can produce unpredictable and excessive packet delays due to the capture effect introduced by the Binary Exponential Backoff (BEB) algorithm. A solution that efficiently overcomes this problem is the PACE algorithm. This paper presents a performance analysis of several schemes of an algorithm based on PACE technology. We developed a mathematical model of the system and we used the Equilibrium Point Analysis (EPA) technique in order to analyze the above model. It is shown that the proposed schemes improve the performance of the network both in terms of throughput and average packet delay.*

## 1. Introduction

In recent years, the demand for networked multimedia and real-time applications has increased the need for higher bandwidth performance on every end-station. With the introduction of Fast Ethernet/100BASE-T technology, the requirement for sufficient bandwidth has been accomplished. 100BASE-T Ethernet operates at 100Mbps and uses the 1-persistent CSMA/CD to control access to a shared medium and the truncated Binary Exponential Backoff (BEB) algorithm for collision resolution.

Even at high speeds, Ethernet networks have some limitations for supporting real time and multimedia traffic, especially due to the well-known capture effect [1]. Several solutions to the capture effect have been proposed in the literature [2]-[6]. Most of these studies have focused on improving network performance by changing the BEB algorithm. However, since the backoff mechanism is implemented in hardware on every end-station, such a solution requires changes in all nodes of the network and is not considered the most appropriate.

PACE technology [1] is a modification to standard Ethernet networks that overcomes the capture effect problem without the drawbacks of solutions based on the modifications of BEB. PACE requires that the changes are made only on the switch or the hub and does not require any modifications at the end-stations. In hub-based networks, the PACE algorithm is used to reduce the access delay and jitter, as compared to the standard Ethernet. In switched Ethernet networks, where the bandwidth issues have been solved, PACE is used to guarantee timely delivery of data and to resolve the problem of traffic prioritisation.

This paper analyses the performance of networks based on the PACE technology and examines how several modifications to the basic PACE algorithm affect this performance. The mathematical model of such a network results to a complicated multi-dimensional Markov chain and cannot be solved using queuing theory. In order to analyse this Markov chain, we utilize an approximate analytic technique called Equilibrium Point Analysis (EPA) proposed by Fukuda and Tasaka [7]-[8]. The EPA can easily analyse a multi-dimensional Markov chain, since it is not necessary to calculate its state transition probabilities.

The remainder of the paper is organized as follows. Section 2 briefly describes the PACE technology and how the capture effect is prevented. In Section 3, we introduce several modifications to the initially proposed PACE algorithm. Section 4 describes the mathematical model used in the performance analysis, while the numerical results of this analysis are presented in Section 5.

## 2. The PACE Algorithm

Ethernet uses the truncated Binary Exponential Backoff algorithm for calculating the retransmission intervals. According to this algorithm, the number of slot times the station has to wait before attempting the n*th* retransmission is a random value uniformly distributed from 0 to $2^n - 1$ time slots, for $0 \le n \le 10$. For the next attempts,

the maximum interval is truncated and remains at its last value. After 16 collisions the packet is rejected.

A result of this algorithm is that after a collision the stations that have suffered more collisions have less probability of acquiring the channel. This asymmetry permits a single busy user to capture the channel for a long period of time, which leads to the capture effect [4]. The major result of this effect is unpredictable and highly variable packet delays that make Ethernet unsuitable for real-time and multimedia applications.

3COM Corp. has developed the PACE technology in order to support real-time and multimedia applications. The original PACE technology has been designed for a switched Ethernet environment and is called *PACE Interactive Access*. A generalization of this technology for the case of a multi-port hub is the *PACE Repeater* algorithm. The major characteristic of the *PACE Repeater* algorithm is that the hub resolves the collisions on the network using the information of the stations attempting to transmit at a given time. The end stations that are connected through the hub implement the standard Ethernet backoff algorithm.

The *PACE Repeater* algorithm works as follows. Upon detection of a collision, the hub selects a winner station from all the active stations. Depending on the winner's station backoff time value, there are three possible transmission scenarios:

- The winner station selects a backoff time value that is earlier than all the others. In this case, no collisions will occur and the winner station successfully transmits its packet.
- The winner station and one or more of the other stations select a backoff time value that is earlier than all the others. In that case, the hub buffers the winner station's packet and retries transmission again at the end of another inter-packet gap (IPG). The retransmission attempt is repeated until a successful transmission occurs.
- One or more of the stations select earlier backoff time values than the winner. In this case, the repeater generates artificial collisions, until the winner station starts transmission.

The selection of the winner station is based on various algorithms like first come, first served (FCFS) or round-robin selection among the stations that have packets to transmit. The result of the above algorithms is that the capture effect is prevented and all the stations have equal access to the network.

## 3. The PACE Algorithm Modifications

In this paper, we follow the basic principles of the PACE technology and we introduce some modifications to the winner selection procedure. Specifically, we consider three different algorithms:

- A random selection scheme, where each active station has equal probability to become the winner station (*algorithm P1*).
- A selection scheme that depends on the value of the backoff counter of the active stations. In this case, a station with higher backoff value has greater probability to become the winner (*algorithm P2*).
- A selection scheme that also depends on the value of the backoff counter of the active stations. In this case, a station with less backoff value has greater probability to become the winner (*algorithm P3*).

In our analysis, we consider end-stations with single buffer capacity and a network of *M* users that satisfies the following assumptions (as defined in [8]):

- The channel propagation delay is identical for all users and it is $\tau$ seconds; thus, the slot duration is $\tau$ seconds.
- The carrier sensing is performed instantaneously.
- Each user generates a packet in a slot with probability $\sigma$. Each packet is of constant length requiring *H* slots for transmission. Therefore, a successful transmission period is of length *(H + 1)* slots.
- When the backoff period has expired, a user senses the channel at the beginning of the next slot. After *J+1* unsuccessful transmission attempts, where *J=15*, the packet is rejected.
- When a collision occurs all users seize their transmissions during the current slot; thus, the length of an unsuccessful transmission period is equal to one slot.
- No transmission errors occur.

Each hub receives all incoming packets and forwards them to all other ports. The hub is capable of measuring the number of collisions that occur in each port after a successful packet transmission; therefore, the hub may create an exact replica of the backoff counter of each station. The hub must use an internal port arbitrator that selects the next winning station by executing the preferred scheduling scheme.

## 4. The Model Analysis

Following the above assumptions, we formulate an approximate model of the system shown in Figure 1. Each user can be in one of the following states:

*T state*: A user with no packet to transmit.

*TR_k state*: A backlogged user with $k$ unsuccessful attempts for transmitting its packet.

*W_k state*: A user that has finished the backoff period after the *k*th unsuccessful attempt.

*$S_{ki}$, $S_{Pi}$ state*: A user that has succeeded in transmitting a packet and will complete the transmission after $i$ slots.

*P state:* A winner user selected by the hub after a collision.

Then, we define the state vector of the Markov chain system as $n \triangleq (n_k, v_k, m_{ki}, m_{Pi}, r : 0 \le k \le J, 0 \le i \le H-1)$,

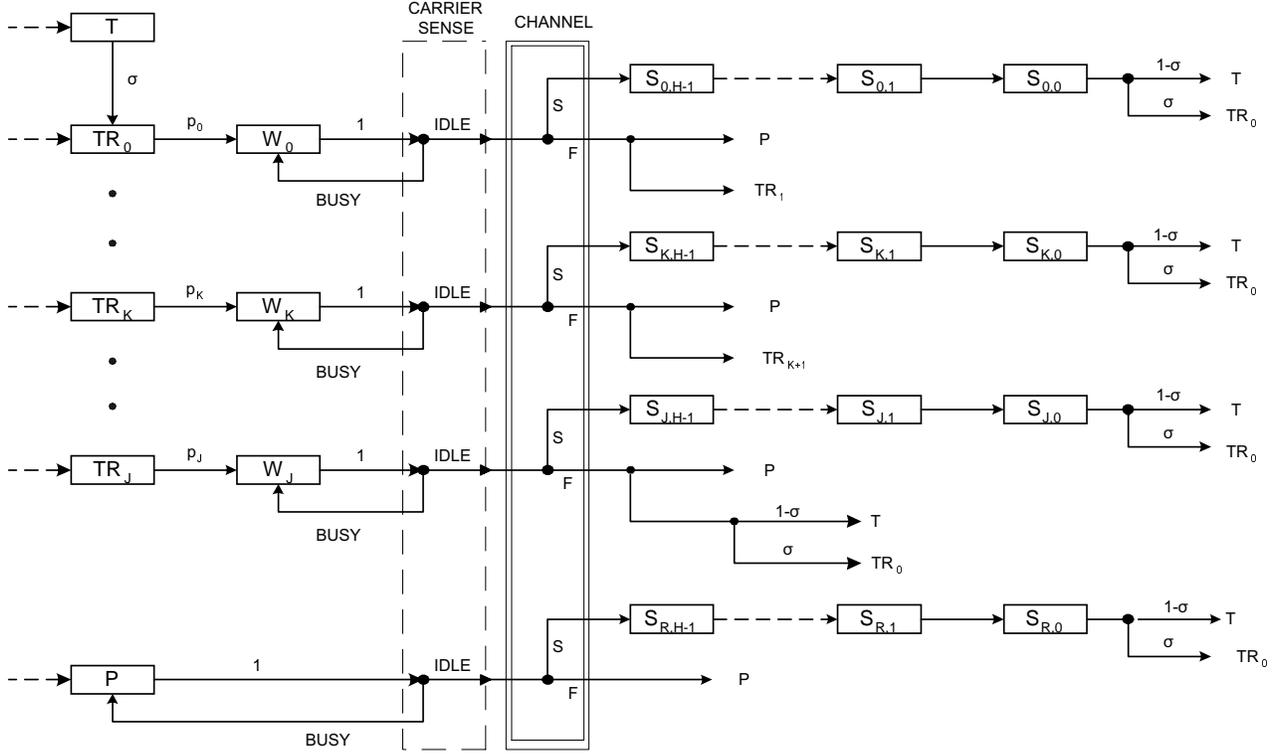**Figure 1. An approximate model of the PACE algorithm.**

where $n_k$ is a random variable representing the number of users in $TR_k$ state, $v_\kappa$ is the number of users in $W_k$ state, $m_{ki}$ is the number of users in $S_{ki}$ state, $m_{Pi}$ is the number of users in $S_{Pi}$ state and $r$ is the number of user in $P$ state.

The EPA method assumes that the system is always at an equilibrium point [7], [8]. By considering that all states are in an equilibrium point, we can get a set of simultaneous equations whose solution gives one or more equilibrium points. In the following analysis, we assume that the system is in steady state $n$.

Let $S_k$ be the conditional probability that a user in $W_k$ state successfully transmits a packet and $S_P$ the conditional probability that a user in $P$ state successfully transmits a packet, given that the channel in state $n$ is idle. Specifically, a user in $W_k$ state successfully transmits a packet when there is no user in state $P$ and one of the following scenarios has occurred:

- The channel was idle at the previous slot and only one user moved from $TR_k$ to $W_k$ at the previous slot.
- There was a collision at the previous slot and only one user moved from $TR_k$ to $W_k$ at the previous slot.
- The channel was busy at the previous slot and only one user moved from $TR_k$ to $W_k$ at the *(H+1)* transmission slots.

Considering that the duration of an unsuccessful transmission period is one slot, the second case can be

regarded as part of the first. Then, the two terms that express the conditional probability $S_k$ are given by:

$$S_{k1} = \left[ n_k p_k \left(1 - p_k\right)^{n_k - 1} \prod_{i=0, i \neq k}^{J} \left(1 - p_i\right)^{n_i} \right] P_I (1 - r),$$
$$\left(0 \leq k \leq J\right) \tag{1}$$

and

$$S_{k2} = \left[ n_k p_k \left(1 - p_k\right)^{n_k - 1} \prod_{i=0, i \neq k}^{J} \left(1 - p_i\right)^{n_i} \left( \prod_{i=0}^{J} \left(1 - p_i\right)^{n_i} \right)^{H} \right] \tag{2}$$
$$\left(1 - P_I\right)(1 - r), \left(0 \leq k \leq J\right)$$

where

$$P_I(n) \triangleq \begin{cases} 1 & \text{if the channel at state } n \text{ is idle} \\ 0 & \text{if the channel at state } n \text{ is busy.} \end{cases} \tag{3}$$

Simplifying the above equations we can express the conditional probability $S_k$ as

$$S_k = (1 - r) n_k p_k \left[ e^{-G} P_I + e^{-(H+1)G} \left(1 - P_I\right) \right],$$
$$\left(0 \leq k \leq J\right) \tag{4}$$

where

$$G \triangleq \sum_{k=0}^{J} n_k p_k. \tag{5}$$

Using the same analysis, we can express the conditional probability $S_P$ as:

$$S_P = r\left[ e^{-G}P_I + e^{-(H+1)G}\left(1-P_I\right)\right]. \qquad (6)$$

After calculating the conditional expectations of the increase in the number of users in each state of Figure 1 in a slot, given that the system is in state $n$, and setting each of them equal to zero [8], we get the following equations:

*T state:*

$$\left\{ M - \sum_{k=0}^{J}\left( n_k + v_k + \sum_{i=0}^{H-1} m_{ki}\right) - \sum_{i=0}^{H-1} m_{Pi} - r\right\}\sigma = (1-\sigma)\sum_{k=0}^{J} m_{0i}$$
$$+(1-\sigma)m_{R0} + (1-\sigma)\left[ rv_J P_I + \left((1-r)v_J - S_J\right)P_I(1-q)\right] \qquad (7)$$

*$W_k$ state:*

$$n_k p_k = v_k P_I \qquad (8)$$

*$TR_k$ state:*

$$n_k p_k = rv_{k-1}P_I + \left((1-r)v_{k-1} - S_{k-1}\right)P_I(1-q) \qquad (9)$$

*$TR_0$ state:*

$$n_0 p_0 = \left\{ M - \sum_{k=0}^{J}\left( n_k + v_k + \sum_{i=0}^{H-1} m_{ki}\right) - \sum_{i=0}^{H-1} m_{Pi} - r\right\}\sigma +$$
$$\sigma\sum_{k=0}^{J} m_{0i} + \sigma m_{R0} + \sigma\left[ rv_J P_I + \left((1-r)v_J - S_J\right)P_I(1-q)\right] \qquad (10)$$

*T state:*

$$rP_I = (r - S_R)P_I + \sum_{k=0}^{J}\left((1-r)v_k - S_k\right)P_I q \qquad (11)$$

*$S_{k0}$ through $S_{k,H-1}$ states:*

$$m_{k0} = m_{k1} = \cdots = m_{k,H-1} = S_k P_I \qquad (12)$$

*$S_{P0}$ through $S_{P,H-1}$ states:*

$$m_{P0} = m_{P1} = \cdots = m_{P,H-1} = S_P P_I \qquad (13)$$

where $q$ is the probability that a user in $W_k$ state is the winner after a collision, given that there is no other user selected previously.

Using a random selection scheme, the probability $q$ is defined as $q = \dfrac{1}{\sum_{k=0}^{J}(n_k + v_k)}$ (*algorithm P1*). When the selection is based on the value of the backoff counter of the active stations the probability is $q_k = \dfrac{k}{\sum_{k=0}^{J}(n_k + v_k)k}$

(*algorithm P2*) or $q_k = \dfrac{15}{\sum_{k=0}^{J}\left[(n_k + v_k)(15-k)\right]}$ (*algorithm*

*P3*) (for $0 \le k \le J$ ).

We also express an equation for the probability $P_I$ using the principles of EPA and the fact that the channel is idle at the beginning of a slot if and only if no user is in any $S_{ki}$ and $S_{Pi}$ state (for $0 \le k \le J$ and $0 \le i \le H-1$). Therefore,

$$P_I = 1 - H\sum_{k=0}^{J} S_k P_I - HS_P P_I. \qquad (14)$$

A solution to the set of simultaneous equations (7)-(14) determines the equilibrium point of the system. Numerically solving the above system of non-linear equations, we determine the number of users in each state and the value of $P_I$ that we utilize in our measurements. The solution is obtained by a non-linear numerical method implemented using a mathematical tool. We used as the method's initial value the exact solution of a similar implementation of standard backoff and then we verified that the solution we obtained was sufficiently accurate. For small values of offered load, the modifications presented in this paper aren't applicable and we consider that the system behaves like a system without the modifications. An explanation for this behaviour is presented in Appendix I.

Now, we can determine the normalized system throughput $S$, defined as the number of successful packet transmissions per $H$ slots (i.e., the transmission time of a packet). According to Figure 1 we have

$$S = \sum_{i=0}^{H-1}\sum_{k=0}^{J} m_{ki} + \sum_{i=0}^{H-1} m_{Pi} \qquad (15)$$

From (12), (13), (4), (6) and (15) the throughput at the equilibrium point is expressed as

$$S = HP_I\left[ r + (1-r)\right]\left[ e^{-G}P_I + e^{-(H+1)G}\left(1-P_I\right)\right]. \qquad (16)$$

We next evaluate the average message delay $D$, which is defined as the average time (in units of $H$ slots) from the moment a packet is generated until its successful transmission is completed. In order to evaluate $D$, we first calculate the average number of total packets in the system, $I$, given by:

$$I = \sum_{k=0}^{J} z_k + z_P \qquad (17)$$

where $z_k$ is the average number of users that have tried $k$ times to transmit their packets and $z_P$ is the average number of users that have been selected by the hub. Since, each user has no more than a packet in his buffer, the number of packets in the system is the same with the number of users in $TR_k$, $W_k$, $S_{ki}$, $P$ and $S_{Pi}$ states (for $0 \le k \le J$ and $1 \le i \le H-1$). Thus

$$z_k = n_k + v_k + (H-1)m_k \qquad (18)$$

and

$$z_P = r + (H-1)m_P \qquad (19)$$

By using the Little's result, we can express the average packet delay (in units of $H$ slots) as:

$$D = \frac{I}{\overline{S}} \qquad (20)$$

where $\overline{S}$ is the expectation of the throughput with respect to $n$. The analytic expression for the average delay is given by:

$$D = \frac{G[1/P_I + (H-1)(1-r)BP_I] + \sum_{k=0}^{J} n_k + r[1+(H-1)BP_I]}{HP_I B[r+(1-r)G]}$$

$$(21)$$

where

$$B = e^{-G}P_I + e^{-(H+1)G}(1-P_I). \qquad (22)$$

## 5. Numerical Results

In this section we use the above analysis in order to compare the performance of the proposed algorithms with the standard Ethernet algorithm. The comparison is based on system throughput and average packet delay results. We also explore the effect of various parameters to the system performance by analysing the system for several values of the number of users $M$ and the packet length $H$. The results concerning the BEB algorithm are obtained using the EPA method to a properly configured system model. The offered load to such a system is $M\sigma H$ (packets per $H$ slots). Although the above analysis is generally applicable to any network data rate, in the rest of this section, we present some results by considering a 100 Mbps Ethernet with 512 bits slot duration.

Figures 2 and 3 show the throughput and the delay curves for a given number of users ($M=25$) and fixed packet size ($H=20$ slots), while Figure 4 shows how the average delay is related to the network throughput. These results show that in all cases of the hub operation presented in this paper yield to better performance when compared to the standard Ethernet. In the case of the random selection scheme, where all the users have the same probability to become the winner station after a collision (*algorithm P1*), the low delay interval is extended for about 30% additional offered load, while the throughput is higher compared to the standard Ethernet.

Furthermore, when the repeater gives priority to the users that have suffered more collisions (*algorithm P2*), there is also improvement but not as much as in *algorithm P1*. This is explained by the fact that in *algorithm P2,* the users with large backoff values have higher transmission probabilities and they attempt retransmissions in larger time intervals. During these intervals, no other station is allowed to transmit. Finally, when the repeater gives priority to the users with smaller backoff values (*algorithm P3*), the network performance is still better than the performance of standard Ethernet and outperforms both
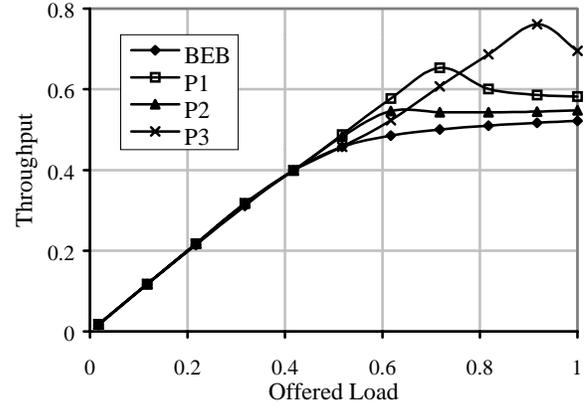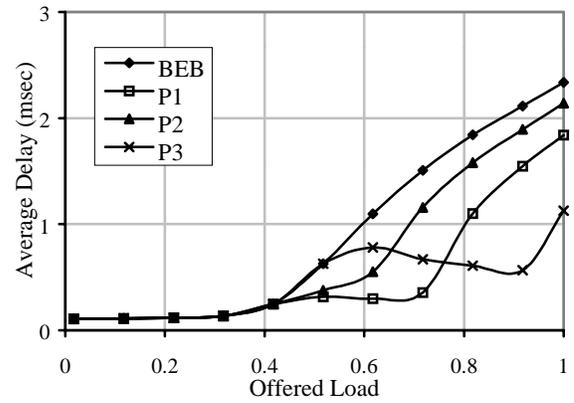


**Figure 2. Throughput versus offered load (H=20, M=25).**



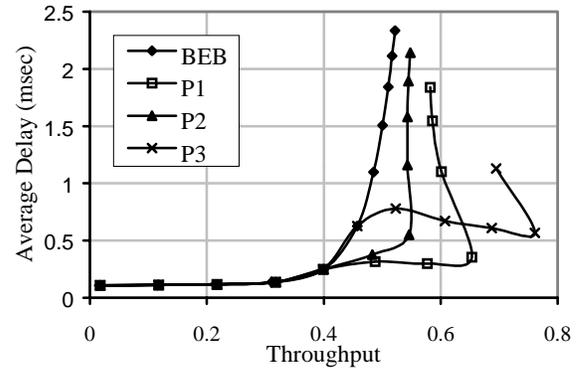**Figure 3. Average packet delay versus offered load (H=20, M=25).**



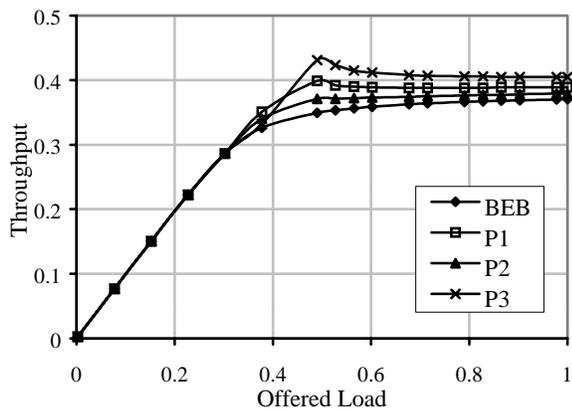**Figure 4. Average packet delay versus throughput (H=20, M=25).**

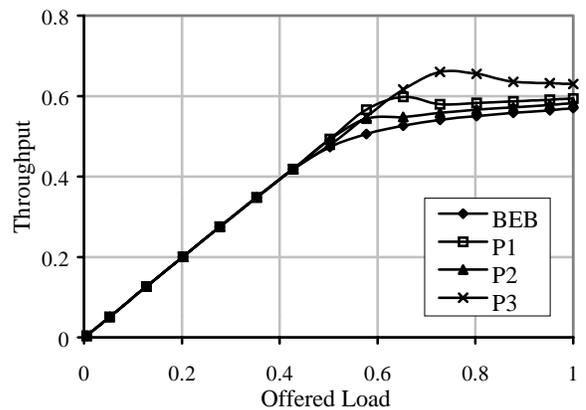**Figure 5. Throughput versus offered load (H=3, M=25).**



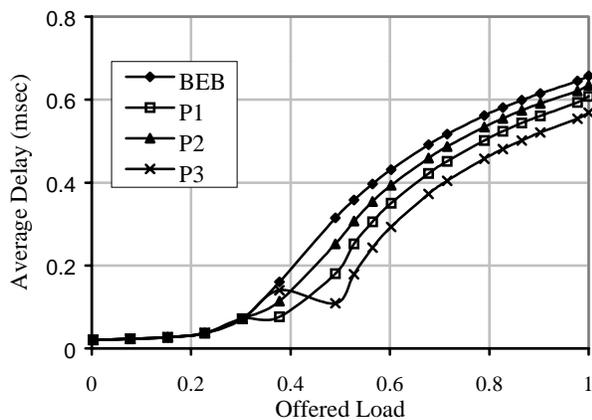**Figure 7. Throughput versus offered load (H=20, M=75).**



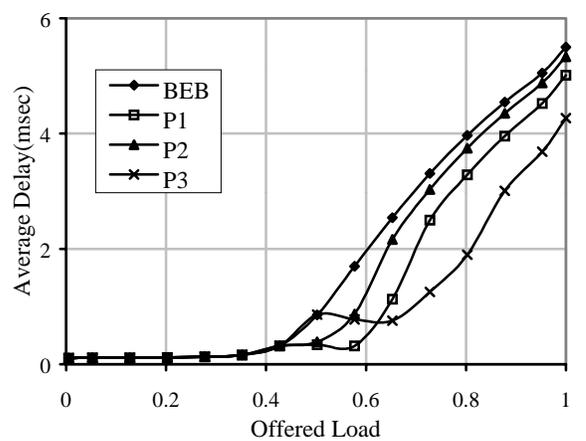**Figure 6. Average packet delay versus offered load (H=3, M=25).**



**Figure 8. Average packet delay versus offered load (H=20, M=75).**

previous cases when the offered load becomes more than 80Mbps.

The same measurements are made for a network with very small packet sizes (*H=3*) and the results are shown in Figures 5 and 6. In this case there is a small improvement in the network delay and throughput due to the high overhead introduced when small packets have to be supported, as explained in [1]. For a real network this is not an issue, since on average the packet sizes are greater than the minimum packet size. Figure 7 and 8 display the same measurements for a larger Ethernet (*M=75*). These results show that all modifications still offer better performance, but in a smaller range due to the increased rate of collisions.

The results obtained from this analytic method illustrate that the modifications on the repeater operation improve the performance of an Ethernet network, especially whenever high offered load has to be supported.

## 6. Conclusions

In this paper we presented the analysis of several modifications to the PACE technology that is applicable to 100 Mbps Ethernet hubs. We developed a Markovian model of the system and we analysed the model using the approximate technique of the equilibrium point analysis (EPA). The presented numerical results indicate that the proposed algorithm modifications improve the performance of the network both in terms of throughput and average packet delay.

# Appendix I.

The method presented in this paper is applicable when the offered load becomes larger than the minimum value that starts causing collisions. Before that value, a station has access to a free channel almost immediately (no collisions) and the system behaviour is identical to a system without the proposed PACE modifications.

In order to determine that value for the offered load, we followed the analysis presented in [9]. The behaviour of every shared multi-access system is defined by observing the response time of the system as a function of the offered load. Figure 9 illustrates the average delay as a function of the offered load for a standard Ethernet network. For small offered load values, there is a slow increase in the average delay and the queuing time for a newly received packet is equal to or less than the time required to receive the next packet. That means that the collisions in the channel are rare. As the load increases, more users want to transmit, causing an increase in the collisions and consequently an increase in the average delay. Therefore, from the delay curve we can determine the maximum value of the offered load that the system can carry without collisions, which we denote as L*. Any additional load beyond that value would increase the rate at which collisions occur.

Figure 10 shows the equilibrium distribution of the number of users according to the value of their backoff counter. For offered load less than L*, the most probable scenario is a user having zero backoff counter. When the load becomes greater than L*, the probability of a user having a non zero backoff counter, which means that this user has made more than one transmission attempts, becomes greater than the probability of a user with zero backoff counter.

From the previous analysis, it is obvious that the necessity of an algorithm that resolves collisions in a multi-access system becomes meaningful when the offered load becomes more than the threshold that depends on the packet length and the number of stations.

# References

[1] 3Com, "PACE Technology: The Realization of Multimedia-Enabled Ethernet". Technical Paper 500676-001, Sept. 1998

[2] T. Gonsalves and F. Tobagi, "On the performance of station locations and access protocol parameters in Ethernet networks," *IEEE Transactions Communications,* vol. COM-36. pp. 441-449, Apr. 1988.

[3] M. L. Molle. "A new binary logarithmic arbitration method for Ethernet," Technical Report CSRI-298, Computer Systems Research Institute, University of Toronto, 1994.

[4] K. K. Ramakrishnan and Henry Yang. "The Ethernet Capture Effect: Analysis and Solution". *Proceedings of IEEE 19th Conference on Local Computer Networks*, Minneapolis Minn, Oct. 1994, pp.228-240.
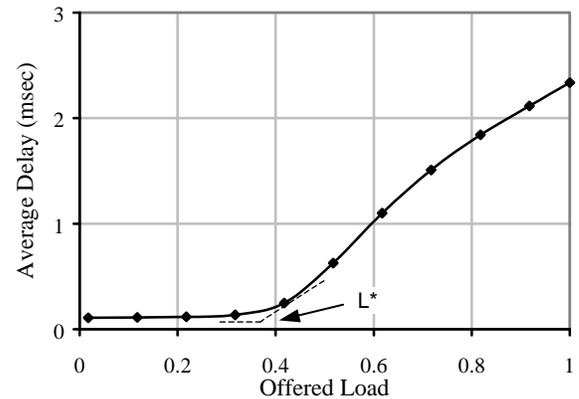
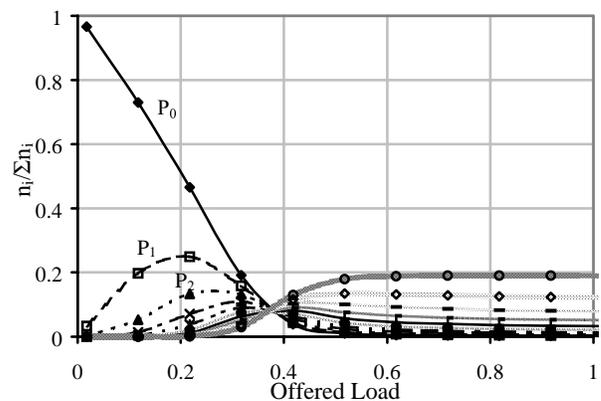**Figure 9. Average packet delay versus offered load for the standard BEB.**



**Figure 10. Distribution of the number of users according to their backoff value.**

[5] W. Hayes and M. L. Molle. "Solving capture in switched two-node Ethernets by changing only one node". *Proccedings of the 20th Annual Conference on Local Computer Networks*, Minneapolis Minn, Oct. 1995, pp.387-395.

[6] K. J. Christensen, "A simulation study of enhanced methods for improving Ethernet performance," *Computer Communications,* 21 (1998) pp. 24-36.

[7] A. Fukuda and S. Tasaka, "The equilibrium point analysis – A unified analytic tool for packet broadcast networks," in *Conf. Rec. GLOBECOM'83,* San Diego, CA, Nov. 1983, pp. 1133-1140.

[8] S. Tasaka, "Dynamic behavior of a CSMA/CD system with finite population of buffered users", *IEEE Transactions on Communications,* vol. COM-34, pp. 576-586, June 1986.

[9] L. Kleinrock, Queueing Systems, Vol. II: Computer Applications. New York: Wiley-Interscience, 1976.