A Probabilistic Analysis of Traffic Separation in Shared Ethernet Using the h-BEB Collision Resolution Algorithm\textsuperscript{1}

Ricardo Moraes, Francisco Vasques
Department of Mechanical Engineering
University of Porto, Portugal
e-mail: \{rmoraes, vasques\}@fe.up.pt

Abstract

Ethernet is a widely used network technology, with a Medium Access Control protocol based on the collision detection between randomly initiated transmissions. It has a non-deterministic behaviour, due to the use of a probabilistic contention resolution algorithm, which impairs the support of real-time communications. In this paper, it is proposed the use of a modified algorithm for the collision resolution in shared Ethernet networks. Such algorithm, referred as high priority Binary Exponential Backoff (h-BEB), provides high priority traffic separation, enabling the support of real-time communications. It allows multiple Ethernet standard devices to coexist with one h-BEB modified station in the same network segment, imposing a higher priority for the transfer of h-BEB related traffic. The probabilistic analysis of traffic separation provided by the proposed algorithm shows that it guarantees an access delay significantly smaller for an h-BEB station, when compared with the access delay for BEB stations.

Plan

1 Introduction
2 The CSMA/CD protocol and the BEB collision resolution algorithm
3 The High priority Binary Exponential Backoff Algorithm
4 Comparative Analysis
5 Review of Relevant Work
6 Conclusions
7 References

Keywords: Ethernet communication, real-time communication.

\textsuperscript{1} This work has been partially supported by IDMEC and by FCT (BD 13203/2003).
1 Introduction

Ethernet is a well-known and extensively used network technology. The first standardized version was approved and released in 1985 as the ANSI/IEEE 802.3 Standard [1]. Its MAC protocol is usually referred as CSMA/CD, which stands for Carrier Sense Multiple Access with Collision Detection. The original ANSI/IEEE 802.3 standard differs from the Ethernet specification, as it describes a whole family of 1-persistent CSMA/CD systems, running at speeds from 1 to 10 Mbps on various media. Commonly, the “Ethernet” label is used to identify all the CSMA/CD protocols, even though it really refers just to one specific implementation of the ANSI/IEEE 802.3 standard.

From 1985, several technological enhancements were covered by multiple extensions to the original standard, clearly improving the Ethernet performance and reliability. The most recent Ethernet version, approved in 2002 as the IEEE specification 802.3ac, is usually referred as the 10 Gigabit Ethernet (or 10 GbE). Previous versions include the Fast Ethernet specification (IEEE 802.3u) that operates at 100 Mbps and the Gigabit Ethernet specification (IEEE 802.3z) that operates at 1 Gbps.

Simplicity was one of the main reasons for the success of Ethernet networks. Such simplicity derives from its MAC protocol, which is based on the collision detection between randomly initiated transmissions. Whenever a collision is detected, a distributed probabilistic algorithm is initiated, to solve the serialization problem between the contending messages. Such algorithm that is based on the local knowledge of occurred collisions, implements a decentralized Medium Access Control Protocol. Nevertheless, one of the main disadvantages is the inherent non-determinism of the probabilistic contention resolution algorithm.

A full-duplex operating mode of Ethernet networks has been introduced in the early 90s (IEEE 802.1D) [2], using bridges (referred as Ethernet Switching Hubs) to interconnect node stations. Such full-duplex operating mode enables the micro-segmentation of the network, by regenerating information only to the receiving port of the bridge, avoiding therefore collisions between messages. Additionally, when using Ethernet Switching Hubs, it is possible to manage network traffic, by means of the adequate setting of data flow permissions and priorities. The network management is specified both by the IEEE 802.1p and the IEEE 802.1q VLAN [3] standards; the latter extends the priority handling
aspects of the 802.1p standard, by providing space in the VLAN Tag to indicate traffic priorities.

Nonetheless, the vast majority of Ethernet networks still operate in heterogeneous environments (Figure 1), with Ethernet Switching Hubs interconnecting both independent node stations and Ethernet Repeater Hubs with multiple interconnected node stations (equivalent to shared Ethernet segments). In such heterogeneous environments, the Switching Hubs impose separate collision domains at each port (network segmentation), allowing the implementation of service policies with different priorities. However, within each of the collision domains (i.e., among node stations interconnected by each Repeater Hub), the network still operates in the traditional shared Ethernet mode; that is, collisions are solved by means of a probabilistic contention resolution algorithm, i.e., the medium access is inherently non-deterministic.

Traditionally, two approaches can be considered to support real-time communications in shared Ethernet environments: avoiding collisions, by controlling the medium access rights of each station (TDMA scheme, token passing, etc.), or ensuring a deterministic collision resolution, by modifying the collision resolution algorithm. A third approach (that is not deterministic) is to reduce the number of occurring collisions, enhancing the network responsiveness to real-time message requests. A brief analysis of the state-of-the-art in shared Ethernet real-time communications is given in Section 5.

Whatever the selected approach, to support real-time communications in shared Ethernet requires the modification of all the interconnected node stations (hardware/software modification at the network adapter level or above). Therefore, it is not possible the coexistence of Ethernet standard devices with modified/enhanced devices in the same shared network.

Figure 1: Heterogeneous Ethernet environment.
segment, which difficult the support of real-time communications (for instance, multimedia video streams) within legacy Ethernet systems.

The major motivation for this paper is to address this issue; that is, to “propose solutions enabling the support of real-time communications in shared Ethernet environments, where Ethernet standard devices can coexist with modified/enhanced devices”.

Basically, we propose the use of a modified algorithm for the collision resolution in shared Ethernet networks. The proposed algorithm, referred as high priority Binary Exponential Backoff (h-BEB), provides high priority traffic separation, enabling the support of real-time communications in shared Ethernet networks. The h-BEB protocol allows multiple Ethernet standard devices to coexist with one h-BEB modified station in the same network segment, imposing a higher priority for the transfer of h-BEB related traffic.

This paper is organized as follows. In Section 2, we recall the traditional BEB collision resolution algorithm used in Ethernet networks, focusing on its performance analysis in a heavily loaded scenario (exact analytical behavior, as described in the literature). In Section 3, the h-BEB algorithm is then presented and analyzed for a similar scenario. It will be shown that, in a shared Ethernet segment, one h-BEB station can coexist with multiple stations implementing the BEB collision resolution algorithm. In Section 4, a comparative analysis is then performed. It will be shown that, for the case of a shared Ethernet segment with one h-BEB station and multiple BEB stations, the h-BEB collision resolution algorithm guarantees an access delay significantly smaller for the h-BEB station, when compared with the access delay for the BEB stations. Finally, a brief analysis of the state-of-the-art in shared Ethernet real-time communications is given in Section 5 and some concluding remarks are done in Section 6.

2 The CSMA/CD protocol and the BEB collision resolution algorithm

2.1 The Binary Exponential Backoff Algorithm - BEB

The CSMA/CD (Carrier Sense Multiple Access with Collision Detection) protocol is the protocol implemented at the MAC layer of both ANSI/IEEE 802.3 [1] and Ethernet local area networks. At this layer,
frames are transferred by the ANSI/IEEE 802.3 and by the Ethernet standards with, respectively, the following format (Figures 2 and 3):

<table>
<thead>
<tr>
<th>7 bytes</th>
<th>PREAMBLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 byte</td>
<td>SFD</td>
</tr>
<tr>
<td>6 bytes</td>
<td>DESTINATION ADDRESS</td>
</tr>
<tr>
<td>6 bytes</td>
<td>SOURCE ADDRESS</td>
</tr>
<tr>
<td>2 bytes</td>
<td>LENGTH</td>
</tr>
<tr>
<td>46-1500 bytes</td>
<td>LLC DATA</td>
</tr>
<tr>
<td>4 bytes</td>
<td>FRAME CHECK SEQUENCE</td>
</tr>
</tbody>
</table>

Figure 2: The 802.3 frame format.

<table>
<thead>
<tr>
<th>8 bytes</th>
<th>PREAMBLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 bytes</td>
<td>DESTINATION ADDRESS</td>
</tr>
<tr>
<td>6 bytes</td>
<td>SOURCE ADDRESS</td>
</tr>
<tr>
<td>2 bytes</td>
<td>TYPE</td>
</tr>
<tr>
<td>46-1500 bytes</td>
<td>DATA</td>
</tr>
<tr>
<td>4 bytes</td>
<td>FRAME CHECK SEQUENCE</td>
</tr>
</tbody>
</table>

Figure 3: The Ethernet frame format.

where, if the DATA field length is smaller than 46 bytes, the PAD field is used to fill out the frame up to its minimum size. For a 10/100 Mbps Ethernet implementation, the following set of parameters is used:

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SlotTime</td>
<td>512 bit times</td>
</tr>
<tr>
<td>InterFrameGap</td>
<td>96 bit times</td>
</tr>
<tr>
<td>AttemptLimit</td>
<td>16</td>
</tr>
<tr>
<td>BackoffLimit</td>
<td>10</td>
</tr>
<tr>
<td>JamSize</td>
<td>32 bit times</td>
</tr>
<tr>
<td>MaxFrameSize</td>
<td>12144 bits</td>
</tr>
<tr>
<td>MinFrameSize</td>
<td>512 bits</td>
</tr>
<tr>
<td>AddressSize</td>
<td>48 bits</td>
</tr>
</tbody>
</table>

Table 1: Ethernet parameters.

Basically, the CSMA/CD protocol works as follows (Figure 4): when a station wants to transmit, it listens to the transmission medium. If the transmission medium is busy, the station waits until it goes idle; otherwise, it transmits immediately. If two or more stations simultaneously begin to transmit, the transmitted frames will collide.

Upon the collision detection, all the transmitting stations will terminate their transmission and send a jamming sequence to ensure that all the transmitting stations recognize the collision and abort the transmission.

When the transmission is aborted due to a collision, it will be repeatedly retried after a randomly evaluated delay (backoff time) until it is, either successfully transmitted, or definitely aborted (after a maximum number of 16 attempts) [1].

One of the key issues is the evaluation of such backoff delay, which is done by locally executing the Binary Exponential Backoff (BEB)

---

² More accurately, when detecting a collision, the station always finishes the transmission of the Preamble and the Start of Frame Delimiter (64 bits), if these have still not been completely transmitted. Afterwards, it transmits a jamming sequence (32 bits), and then stops.
algorithm. Such algorithm operates as follows: after the end of the jamming sequence, the time is divided into discrete slots, whose length is equal to the slot time\(^3\). The backoff time is given by \( t_{\text{backoff}} = r \times T \), where \( r \) is a random integer in the range \( 0 \leq r \leq 2^k - 1 \), \( k \) is the smaller of \( n \) or 10 (\( n \) is the number of retransmission attempts) and \( T \) is the slot time in seconds. This means that the station will wait between 0 and \( 2^n - 1 \) slot times, being \( n \) the number of collision resolution rounds. Finally, after 10 attempts, the waiting interval is fixed at 1023 slot times, and after 16 attempts a failure is reported and the transmission is aborted.

The CSMA/CD protocol seems to have a random queue service discipline, \( i.e. \), the message to be transferred after a successful transmission seems to be randomly chosen among the \( N \) hosts with ready messages. However, Christensen [4] demonstrated that the BEB algorithm imposes a last come first serve policy, as a station with the more recently queued packet, will have an higher probability for the acquisition of the medium.

Another particularity of the CSMA/CD protocol is the **Packet Starvation Effect**. Wheten et al. [5] demonstrated that, in heavily loaded networks,

\(^3\) For Ethernet and Fast Ethernet (10/100 Mbps) networks, one slot time is the time required for transmitting the minimum frame size (512 bits), that is, respectively, 51.2 and 5.12 \( \mu \text{sec} \). For Gigabit Ethernet (1Gbps), one slot time corresponds to the transmission time of 4096 bits.
an older packet will have a smaller probability to be transferred than a newer one. For example: consider that 2 stations have packets ready to be transmitted (station1 and station2), which will be transmitted at approximately the same time; a collision will occur and then both stations will backoff during a randomly selected delay between 0 and $2^n-1$ slot times, where $n$ is the number of previous collisions. In the first collision resolution interval, if station1 waits 0 slot times and station2 waits 1 slot time, station1 will transmit its packet while station2 will wait. Supposing that station1 has other packets to be transferred, then, in the following collision the backoff time of station1 will be 0 or 1, and the backoff time of the station2 will be 0, 1, 2 or 3. Therefore, station1 will have a higher transmission probability. Such Packet Starvation Effect will occur whenever a station has a sequence of packets to be consecutively transferred, if the network interface adapter is able to effectively contend for the network access at the end of every transmitted frame. Otherwise, one other station will acquire the transmission medium.

### 2.2 Analytical Study of the BEB Algorithm

One of the first Ethernet performance analysis was presented in [6], where the authors draw up a set of formulas to execute the exact analysis in heavily loaded Ethernet networks. In that analysis, a constant retransmission probability on each slot has been assumed, and the successful retransmission probability (on the next slot) has been considered to be equal to a constant: $p$. Therefore, for the case of $K$ active hosts (hosts with packets ready to be transmitted), the probability that only one host will transmit in the beginning of a slot (thus avoiding a collision) is [6]:

$$A = K \times p \times (1 - p)^{K-1}$$

Such probability $A$ is maximized when $p=1/K$. (equal probability of successful retransmission). Such assumption is an interesting approximation for the real backoff function, as has been shown in multiple simulation studies (e.g. [7] [8]). Thus,

$$A = (1 - \frac{1}{K})^{K-1}$$

The probability that a host will wait during just 1 slot is $A(1-A)$, while the probability that the contention interval will be exactly $n$ slots is:

$$P_n = A \times (1 - A)^{n-1} \quad n\geq1$$
The estimated number of stations trying to transmit is truncated to 1023. Truncating imposes an upper bound to the time interval (backoff delay) that any station must wait before trying to transmit again. Therefore, it results on an upper bound of 1024 potential slots for transmission. Such upper bound imposes a maximum number of 1024 stations that can be supported by a half duplex Ethernet system [9].

The average number of contention slots is given by [6]:

\[
Z = \sum_{n=0}^{\infty} n \times A \times (1 - A)^n = \frac{1 - A}{A}
\] (4)

Considering \(P\) as the packet length (expressed in bits) and \(C\) as the network data rate (expressed in bps), the ratio \(P/C\) represents the transmission time of an average packet (expressed in seconds). Therefore, the channel efficiency \(E\) (time during which packets are being effectively transmitted) can be evaluated as the ratio between the transmission time and the transmission plus contention intervals:

\[
E = \frac{P/C}{P/C + (Z \times T)}
\] (5)

where \(Z \times T\) represents the average acquisition time before effectively transmitting (\(T\) is the slot time in seconds). Figure 5 illustrates the “channel efficiency” in heavily loaded networks, assuming a 10Mbps Ethernet network (\(C=10\) Mbps; \(T=51.2\) µs).

According to Boggs et al. [10], one of the most widely accepted Ethernet myths is that it saturates at an offered load of 37%. Such assertion is well founded when dealing with short sized frames and a significant number of hosts. However, for longer frames, the channel efficiency is
significantly improved. Schoch and Hupp [11] presented measurements results indicating that for 4096 bit frames and small number of hosts, the channel utilization approaches 97%; however, for small packets and larger number of hosts the utilization approaches 1/e, that is, approaches the 37% bound. These results are consistent with the Metcalfe and Boggs analysis [6], as can be depicted from the channel efficiency results represented in Figure 5.

3 The High priority Binary Exponential Backoff Algorithm

3.1 Rationale

Within the traditional CSMA/CD protocol, the BEB algorithm delays the frame retransmission during a time interval (backoff delay) that is a probabilistic function of the number of previous collisions. This means that the retransmission probability does not depend on the type of traffic, but just on the state of the collision counter of each particular station.

As a consequence, in CSMA/CD networks it is not possible to provide traffic separation at the MAC level on a data stream basis. To provide such traffic separation, which is a requirement to support real-time communications, there are usually two different approaches: either avoiding collisions, by controlling the medium access rights of each station (TDMA scheme, token passing, etc.), or ensuring a deterministic collision resolution, by modifying the collision resolution algorithm. A third approach (that is not deterministic) is to reduce the number of occurring collisions, enhancing the network responsiveness to real-time message requests.

The drawback of such traditional approaches is that they do not allow the coexistence of Ethernet standard devices together with modified devices in the same network segment, which means that legacy shared Ethernet systems cannot support real-time communications without extensive modifications. A brief analysis of the state-of-the-art in shared Ethernet real-time communications is given in Section 5.

To address this problem, we propose the use of a modified algorithm, referred as “high priority Binary Exponential Backoff (h-BEB)”, for the collision resolution in shared Ethernet networks, which allows Ethernet standard devices to coexist with one h-BEB modified station, imposing a higher priority for the transfer of h-BEB related traffic.
3.2 The h-BEB Algorithm

A station implementing the h-BEB algorithm operates as follows: it has the same operating behavior of a CSMA/CD station, except in what concerns the evaluation of the backoff delay. Whenever there is a collision, the station starts immediately transmitting (backoff interval equal to zero) after the end of the jamming sequence.

This behavior guarantees the highest transmitting probability to the h-BEB station in a shared Ethernet segment with multiple BEB stations. The h-BEB station will always try to transmit its frame in the first slot, while all the other stations implementing the BEB algorithm will wait between 0 and $2^n-1$ slot times, where $n$ is the number of collision resolution rounds. Therefore, the h-BEB modification consists just in setting the backoff delay parameter to 0 in the h-BEB station. Figure 6 summarizes the dynamic behavior of a station implementing the h-BEB collision resolution algorithm.

The h-BEB collision resolution algorithm can be used to support real-time traffic separation, as the traffic generated by the h-BEB station will be always transferred prior to the traffic generated by the other stations. Therefore, this algorithm is adequate to support real-time communications in shared Ethernet segments. This behavior is highly adequate to, for instance, real-time video/voice transferring applications in legacy shared Ethernet networks. By simply plugging a notebook computer with the modified hardware to the network, it becomes possible
to transfer traffic at a higher priority than the traffic generated by all the other stations.

3.3 Shortcomings of the Current Version of the h-BEB Algorithm

A shortcoming of the current version of the h-BEB algorithm is that it is adequate just for shared Ethernet segments with at most one h-BEB station. However, such limitation can be easily overcome, implementing a token passing procedure among the multiple h-BEB stations in the network segment. By controlling the token holding time of each h-BEB station, it is possible to guarantee the real-time behavior of the supported applications.

Nevertheless, it must be clear that, in spite of this limitation, the proposed algorithm accomplishes its target, as it allows Ethernet standard devices to coexist with one h-BEB station in the same network segment, where the h-BEB traffic has higher priority than the traffic generated by the BEB stations.

3.4 Analytical Study of the h-BEB Algorithm

One of the assumptions of the performance analysis carried out by Metcalfe and Boggs [6] is that each station transmits with an equal probability \( p = 1/K \). Obviously, such assumption is not suitable for the analysis of the h-BEB algorithm, as in the h-BEB case one of the stations (the privileged station) transmits at a higher probability. Therefore, new and adequate formulae must be devised to perform the probabilistic analysis of the h-BEB collision resolution algorithm. Following, in this section, we derive an exact probabilistic analysis for the h-BEB collision algorithm, by sequentially analyzing 2, 3 and 4 network station scenarios.

Firstly, consider the case of 2 stations connected to a shared Ethernet network, station A and station B implementing, respectively, the h-BEB and the BEB algorithms. Following an initial collision, the probability of station A winning the collision resolution is computed as follows: in the first collision \( (n=1) \) the probability of station A to transmit is 1/2, as station A will always try to transmit in slot 0, while station B will backoff during 0 or 1 slot times (according to the BEB algorithm). In the case of a second collision \( (n=2) \), station B will wait 0, 1, 2 or 3 slot times, while station A will try to transmit in slot 0 again, but now with a transmission probability of 3/4 (the probability of a new collision is just 1/4).

Therefore, the probability that station A wins the collision resolution is:
\[ P(n,1) = \frac{2^n - 1}{2^n} = 1 - 1 \times 2^{-n} \]  

(6)

where \( n \) represents the number of collision rounds and “1” represents the number of BEB stations in the network segment.

Consider now the case of 3 stations connected to the shared Ethernet network: station A with h-BEB algorithm and stations B and C implementing the BEB algorithm. The probability of station A winning the collision resolution is computed as follows: in the first collision \((n=1)\) the station A try to transmit in slot 0, while station B and C will backoff during 0 or 1 slot times (according to the BEB algorithm). Thus, the possible backoff numbers for the three stations are:

\[
\begin{array}{cccc}
0,0,0 & 0,0,1 \\
0,1,0 & 0,1,1 & \text{Station A wins}
\end{array}
\]

and the probability of station A winning is 1/4. In the case of a second collision \((n=2)\), stations B and C will wait 0, 1, 2 or 3 slot times, while station A will try to transmit in slot 0 again. Thus, the possible backoff numbers are:

\[
\begin{array}{cccccccc}
0,0,0 & 0,1,0 & 0,2,0 & 0,3,0 \\
0,0,1 & 0,1,1 & 0,2,1 & 0,3,1 \\
0,0,2 & 0,1,2 & 0,2,2 & 0,3,2 \\
0,0,3 & 0,1,3 & 0,2,3 & 0,3,3 & \text{Station A wins}
\end{array}
\]

and the probability of station A winning is 9/16. In the case of a third collision \((n=3)\), stations B and C will wait 0, 1, 2, 3, 4, 5, 6 or 7 slot times, while station A will try to transmit in slot 0 again. Thus, the possible backoff numbers are:

\[
\begin{array}{cccccccccccc}
0,0,0 & 0,1,0 & 0,2,0 & 0,3,0 & 0,4,0 & 0,5,0 & 0,6,0 & 0,7,0 \\
0,0,1 & 0,1,1 & 0,2,1 & 0,3,1 & 0,4,1 & 0,5,1 & 0,6,1 & 0,7,1 \\
0,0,2 & 0,1,2 & 0,2,2 & 0,3,2 & 0,4,2 & 0,5,2 & 0,6,2 & 0,7,2 \\
0,0,3 & 0,1,3 & 0,2,3 & 0,3,3 & 0,4,3 & 0,5,3 & 0,6,3 & 0,7,3 \\
0,0,4 & 0,1,4 & 0,2,4 & 0,3,4 & 0,4,4 & 0,5,4 & 0,6,4 & 0,7,4 & \text{Station A wins} \\
0,0,5 & 0,1,5 & 0,2,5 & 0,3,5 & 0,4,5 & 0,5,5 & 0,6,5 & 0,7,5 \\
0,0,6 & 0,1,6 & 0,2,6 & 0,3,6 & 0,4,6 & 0,5,6 & 0,6,6 & 0,7,6 \\
0,0,7 & 0,1,7 & 0,2,7 & 0,3,7 & 0,4,7 & 0,5,7 & 0,6,7 & 0,7,7
\end{array}
\]
and the probability of station A winning is now 49/64. Therefore, the
probability that station A wins the collision resolution is:

\[ P(n,2) = \frac{2^{2n} - 2^{n+1} + 1}{2^{2n}} = 1 - 2 \times 2^{-n} + 1 \times 2^{-2n} \]  

Applying the same reasoning for the collision resolution in a network
with one h-BEB station and 3 BEB stations, the possible backoff
numbers for the first collision \(n=1\) are:

\[
\begin{align*}
0,0,0,0 & 0,0,0,1 \\
0,0,1,0 & 0,0,1,1 \\
0,1,0,0 & 0,1,0,1 \\
0,1,1,0 & 0,1,1,1
\end{align*}
\]

and the probability of station A winning is 1/8. In the case of a second
collision \(n=2\), stations B, C and D will wait 0, 1, 2 or 3 slot times, while
station A will try to transmit in slot 0 again. Thus, the possible backoff
numbers are:

\[
\begin{align*}
0,0,0,1 & 0,0,1,0 & 0,0,2,0 & 0,0,3,0 & 0,1,0,0 & 0,1,1,0 & 0,1,2,0 & 0,1,3,0 \\
0,0,0,2 & 0,0,1,1 & 0,0,2,1 & 0,0,3,1 & 0,1,0,1 & 0,1,1,1 & 0,1,2,1 & 0,1,3,1 \\
0,0,0,3 & 0,0,1,2 & 0,0,2,2 & 0,0,3,2 & 0,1,0,2 & 0,1,1,2 & 0,1,2,2 & 0,1,3,2 \\
0,0,0,4 & 0,0,1,3 & 0,0,2,3 & 0,0,3,3 & 0,1,0,3 & 0,1,1,3 & 0,1,2,3 & 0,1,3,3 \\
0,2,0,0 & 0,2,1,0 & 0,2,2,0 & 0,2,3,0 & 0,3,0,0 & 0,3,1,0 & 0,3,2,0 & 0,3,3,0 \\
0,2,0,1 & 0,2,1,1 & 0,2,2,1 & 0,2,3,1 & 0,3,0,1 & 0,3,1,1 & 0,3,2,1 & 0,3,3,1 \\
0,2,0,2 & 0,2,1,2 & 0,2,2,2 & 0,2,3,2 & 0,3,0,2 & 0,3,1,2 & 0,3,2,2 & 0,3,3,2 \\
0,2,0,3 & 0,2,1,3 & 0,2,2,3 & 0,2,3,3 & 0,3,0,3 & 0,3,1,3 & 0,3,2,3 & 0,3,3,3
\end{align*}
\]

and, similarly to the previous cases, the probability of station A winning
is 27/64. In the case of a third collision \(n=3\), stations B, C and D will
wait 0, 1, 2, 3, 4, 5, 6 or 7 slot times, while station A will try to transmit
in slot 0 again. The same reasoning for this case could be applying and
the probability of station A winning is equal to 343/512.

Therefore, the probability of the h-BEB station to win the collision
resolution in \(n\) collision rounds is:
\[
P(n,3) = \frac{2^{3n} - [2^{2n} + 2^n \times (2^n - 1) + (2^n - 1) \times (2^n - 1)]}{2^{3n}}
\]

\[
P(n,3) = 1 - 3 \times 2^{-n} + 3 \times 2^{-2n} - 1 \times 2^{-3n}
\]  

(8)

Applying the same reasoning to the 4 and 5 BEB stations scenarios, the probability of the h-BEB station to win the collision resolution in \( n \) rounds would be:

\[
P(n,4) = 1 - 4 \times 2^{-n} + 6 \times 2^{-2n} - 4 \times 2^{-3n} + 1 \times 2^{-4n}
\]

\[
P(n,5) = 1 - 5 \times 2^{-n} + 10 \times 2^{-2n} - 10 \times 2^{-3n} + 5 \times 2^{-4n} - 1 \times 2^{-5n}
\]  

(9) \hspace{1cm} (10)

Observing equations (7), (8), (9), (10) and (11), it is clear that the equation coefficients of \( P(n,N) \) make up a Pascal Triangle, as follows:

\[
\begin{array}{c c c c c c}
1 & 1 \\
1 & 2 & 1 \\
1 & 3 & 3 & 1 \\
1 & 4 & 6 & 4 & 1 \\
1 & 5 & 10 & 10 & 5 & 1 \\
\end{array}
\]  

(11)

Therefore, the general expression for \( P(n,N) \) is given by:

\[
P(n, N) = \sum_{j=0}^{N} (-1)^j \binom{N}{j} \times 2^{-jn}
\]

(12)

where the coefficients of Pascal Triangle are given by:

\[
\binom{N}{j} = \frac{N!}{j!(N-j)!}
\]

(13)

and \( N \) is the number of BEB stations in the network (\( N+1 \) is the total number of stations).

The probability analysis devised in this section addresses a rarely occurring case, as it is based on the assumption that, at the start of the transmission attempt, all the network stations participate in the contention process (heavily loaded network scenario). For more realistic load scenarios, at our best knowledge, it is not possible to perform an exact analysis, similar to the above-presented one. For such case, the performance analysis must be done by simulation, and therefore is not exact.
Nevertheless, the analysis for the heavily loaded network scenario provides valuable information that cannot be obtained from the simulation analysis. First of all, it provides exact results whether the simulations analysis just provides approximate results. Then, it provides also the type of results that are useful for the worst-case analysis, as such results are based on worst-case assumptions.

4 Comparative Analysis

Multiple mathematical models have been developed to represent the behavior of standard Ethernet networks; however, due to the inherent complexity of the mathematical models, several abstractions are usually made. Boggs et al. [10] present an interesting survey of performance studies addressing Ethernet standard networks. Other relevant works can be also referred [12] [13] [14] [15].

![Figure 7: Comparative analysis scenarios.](image)

In this section we present some preliminary results for the performance analysis of the h-BEB collision resolution algorithm, and compare them with results obtained from the traditional BEB algorithm. The comparative analysis model considers a 10 Mbps Ethernet network, where multiple stations are interconnected with a special station (Figure 7) implementing either the h-BEB (enhanced Ethernet mode) or the BEB algorithms (traditional Ethernet mode).
The performed analysis compares the results obtained from Equation (3) (according to the Metcalfe and Boggs analysis) with the results obtained from Equation (12). In order to have comparable results, let us analyze again both equations.

Equation (3) indicates the probability that the contention interval will be exactly $n$ slots, when all the stations implement the BEB collision resolution algorithm (scenario 2). Therefore, for this case, the probability that the special station will win the collision resolution in $n$ collision rounds is:

$$P(n, N) = \frac{A \times (1 - A)^{n-1}}{N + 1} \quad n \geq 1$$  \hspace{1cm} (14)$$

where $N+1$ is the total number of stations in the network segment ($N$ generic stations plus one special station), as all the stations have equal probability when accessing the communication medium.

On the other case (scenario 1), the probability that the h-BEB station (special station) will win the collision resolution in $n$ collision rounds is directly given by Equation (12). Therefore, the results obtained from Equation (14) can be directly compared with those obtained from Equation (12).

Two sets of results are analyzed: In the first set, it is represented the probability of transmission for the special station after $n$ collision resolution rounds, for both the traditional and the enhanced Ethernet modes. That is, results obtained from Equation (14) are being directly compared with results obtained from Equation (12). Figures 8, 9, 10 and 11 illustrate the results of such analysis. It becomes clear that, the special station has a much higher transmission probability in the enhanced Ethernet mode than in the traditional mode (as it was expected).

In the second set of results, it is represented the network accessibility, that is, the probability that the contention interval will be exactly $n$ slots. In this case, results from Equation (3) are directly compared with results obtained from Equation (12), as while in the traditional Ethernet mode any station can access the communication medium, in the enhanced Ethernet mode, the h-BEB station will always win the contention. Therefore, in the enhanced Ethernet mode, the probability that the h-BEB station will be able to access the communication medium after $n$ collision resolution rounds is equal to the probability that the contention interval will be exactly $n$ slots. Figures 12, 13, 14 and 15 illustrate the network accessibility for both the traditional and the enhanced Ethernet modes.
From these results, it becomes clear that in the enhanced Ethernet mode, the network accessibility is smaller than in the traditional mode, for the initial collision resolution rounds. These are expected results, as in the enhanced mode, the special station does not allow any other station to transmit, while it has not succeed to transfer its packets. Therefore, the contention period will be longer than in the traditional mode, whenever the special station has packets to be transferred.
5 Review of Relevant Work

Two approaches can be considered to support real-time communications in shared Ethernet environments: either avoiding collisions, by controlling the medium access rights of each station (TDMA scheme, token passing, etc.), or ensuring a deterministic collision resolution scheme, by modifying the collision resolution algorithm. A third approach (that is not deterministic) is to reduce the number of occurring collisions, enhancing the network responsiveness to real-time message requests. Figure 16 illustrates the three approaches.
Controlling the Medium Access Rights

- TDMA Token Passing
- FTT-Ethernet
- VTPE

Imposing a Deterministic Collision Resolution scheme

- CSMA/DCR
- DOD-CSMA-CD

Reducing the Number of Occurring Collisions

- Virtual time CSMA
- Window Protocols
- Dynamic pi-persistent
- CABEB
- BLAM
- Traffic Smoothing

Figure 16: Supporting Real-Time Communication in shared Ethernet Networks.

5.1 Controlling the Medium Access Rights

Presently, most part of the solutions to avoid collisions in Ethernet networks are based on the Switched Ethernet standard IEEE 802.1D [2], combined with the prioritizing mechanisms defined by the IEEE standard 802.1p. These approaches are out of the scope of this brief survey, as they do not address shared Ethernet environments.

One of the first solutions to eliminate collisions in shared Ethernet environments was proposed by Chen and Lu [16] based on the TDMA (Time Division Multiple Access) paradigm, where each station has a pre-allocated transmission time interval. Another approach to eliminate collisions has been proposed by Pritty et. al. [35]. Such approach is based on the use of the Timed Packet Release principle, where a Monitor Node periodically transmits a Slot Pulse to synchronize the medium access. More recently, Pedreiras and Almeida proposed the use of the FTT (Flexible Time-Triggered) paradigm to schedule communications in a shared Ethernet network [17]. In such approach, time is divided in synchronous and asynchronous windows, which are used to, respectively, statically schedule the hard real-time traffic and dynamically serve the soft real-time requests.

Another approach to provide a deterministic collision-free environment is to use a token passing procedure, where each station is allowed to access the medium only during the token holding intervals. Venkatramani and Chiueh [18] proposed the RETHER (real-time Ethernet) protocol, where the network is initialized in the CSMA mode until a real-time request arrives, passing then to the RETHER mode, where all nodes operate according to a specified token passing protocol. In [19], J. Lee et al.
proposed the use of the IEEE 802.4 Token-Passing Bus Access method directly on top of the Ethernet Physical Layer, where a specifically proposed service translator performs the required translation of frame formats and interface functions. More recently, F. Carreiro et al. [20] proposed the use of the VTPE (Virtual Token-Passing Ethernet) procedure, to ensure a deterministic collision-free environment. This procedure is based on the common knowledge of time and also on the state of the access counter of each network node, implementing an implicit (virtual) token passing among network nodes.

5.2 Imposing a Deterministic Collision Resolution scheme

Another approach to support real-time communications in shared Ethernet environments is to impose a deterministic collision resolution scheme, ensuring that the colliding frames are serialized in an upper-bounded time interval.

One of the first proposals was done by Takagi et al. [21], which proposed a CSMA/CD protocol with deterministic contention resolution (DCR). In the absence of collisions, the CSMA/DCR protocol implements the traditional CSMA/CD access method and, when in collision situations, a binary search tree is used to sort the colliding nodes; a priority hierarchy is enforced, i.e., higher priorities nodes try to access the transmission medium prior to the lower priority nodes, using an implicit token passing mechanism. The DOD-CSMA-CD protocol [22] improved the CSMA-DCR protocol, as it uses network station indices that are computed online, rather than pre-assigned ones.

5.3 Reducing the Number of Occurring Collisions

Finally, the third approach to support real-time communications in shared Ethernet environments is to reduce the number of occurring collisions, which directly enhances the network responsiveness to real-time message requests. Note however that the proposed solution is still non-deterministic, as collisions are still solved in a probabilistic way.

Molle and Kleinrock [23] proposed a CSMA algorithm, called Virtual Time CSMA (VTCSMA), that uses a probabilistic approach combined with specific timing parameters (arrival time, laxity, deadline, length) for the collision resolution, enabling the implementation of different scheduling policies. Zhao and Ramamritham [24] presented a performance analysis of the four VTCSMA protocols: VTCSMA-A, VTCSMA-T, VTCSMA-D and VTCSMA-L, which implement the
minimum-arrival-time-first, minimum-transmission-time-first, minimum-deadline-first, and minimum-laxity-first policies, respectively.

Another relevant modification proposed to the CSMA/CD protocol is the Window Protocol [25] [26] [27] [28], which implements a dynamic time window to reduce the number of occurring collisions. It operates as follows: when just one host has a message ready to be transmitted, if the message is within the window, then it will be sent; if several hosts have messages to be transmitted within the window, the window size is reduced according to the selected policy, until there is just one remaining message within the window; if there are no nodes with messages within the window, the window size can be increased.

In [29] the authors presented a modified CSMA/CD protocol, called the Dynamic pi-persistent CSMA/CD protocol. It is similar to the p-persistent CSMA/CD protocol, but with a transmission probability that depends on the laxity of the ready packet. It also implements a time window to reduce the number of collisions in heavily loaded systems.

Molle et al. [12, 30] proposed a BEB compatible algorithm, the Binary Logarithmic Arbitration Method (BLAM), with a modified collision counter policy. According to Christensen [4], following a successful transmission, all the stations will have an equal access probability to the medium. Therefore, it eliminates the packet starvation effect [5]. The Capture Avoidance Binary Exponential Backoff (CABEB) algorithm proposed by Ramakrishnan and Yang [31] addresses also the packet starvation effect. It enhances the collision resolution algorithm for the special case when a station attempts to capture the channel following an uninterrupted sequence of message transfers. Another alternative for calculating the backoff has been proposed by Pritty [36], where the evaluation of the backoff delay at each station is adapted to the local traffic characteristics (real-time data and video streams).

Finally, another approach is to use the traffic smoothing mechanism, introduced by Kweon et al. [32], where the packet generation rate (from the upper layers) is kept below a defined threshold, called the network-wide input limit. Several policies for the traffic smoothing have been proposed: the HMD (Harmonic-Increase and Multiplicative Decrease) [28] uses the credit bucket depth and the refresh period as dynamic traffic regulator; in the absence of collisions, it periodically increases the input bound through periodically reducing the refresh period. In [33] the smoothing actions are performed by a fuzzy controller, where the network load is observed along determined time intervals, via the
throughput measurement and the number of occurring collisions. Finally, in [34] a middleware system is proposed to regulate the network access by means of a polling mechanism.

6 Conclusions

This paper describes the traditional BEB collision resolution algorithm used in Ethernet networks, focusing on its performance analysis in heavily loaded networks. It proposes the use of a modified algorithm: the h-BEB algorithm, for the collision resolution in shared Ethernet networks. The probabilistic analysis of the h-BEB algorithm is performed for a heavily loaded network scenario. The analysis results show that an h-BEB station has a significantly higher probability to send a message up to the \(i^{th}\) collision round than any BEB station. Therefore, the h-BEB collision algorithm is highly adequate to support real-time communications in legacy shared Ethernet networks.

References


