ETHERNET TRANSMISSION DELAY DISTRIBUTION:
AN ANALYTIC MODEL

Kiyoshi YONEDA

Tokyo, Japan

ABSTRACT
An analytic model is presented for
calculating Ethernet transmission delay
distribution, not confined to a few of its
representative values. An ether state model
allowing arbitrary packet size distribution
is obtained on the way, which yields a closed
form throughput formula. The model is
elementary and straightforward, using a
discrete Markov chain and standard
manipulation of distributions. Comparison
with a more complex ether state model shows
almost identical throughput characteristics
in the range of practical interest. A sample
computation is compared with a simulation of
one bit accuracy.

1. INTRODUCTION
How real time is Ethernet? An effort to
answer this boils down to finding the ether
(or channel) acquisition delay, viz. the time
required to start a successful packet
transmission after the packet is passed to
the transceiver. Since Ethernet makes use of
randomness, the delay tends to incorporate a
large variability. While most users are
satisfied to know the mean delay, some demand
the coefficient of variation, while others
want to know quantiles. For instance, the
99% point in delay distribution is sometimes
used as a measure of acoustic quality in
packet voice communication; 99.7% delay seems
to be a customary index in process control.
All such performance indices may be produced
if the entire delay distribution is known.
The distribution has traditionally been made
available through discrete event simulation:
in order to observe, say, three packets
exceeding 99.7% delay, a sample of size one
thousand is required on the average.

An analytic model to compute the ether
acquisition delay distribution is provided in
this paper, not confined to just a few of
distribution's representative values. As a
byproduct, a simple model of the ether state
is introduced which allows arbitrary packet
size distributions; its solution gives a
closed form throughput formula.

10[Mb/s] Ethernet specifications are
provided in [1]. Its data link layer
protocol is summarized in Figure 1 for
convenience. This protocol is a well-known
variation in a 1-persistent carrier sense
multiple access with collision detection
(CSMA/CD). A somewhat more suggestive term,
persistent listen while talk (PLWT) [2], will
be preferred throughout. "Persistent" refers
to the property where a deferred station
starts talking as soon as the present talker
is over. Note that, in case there are two or
more stations in deference, a collision will
inevitably take place at the time the talking
station goes quiet and all those in
deference begin to talk.

The backoff strategy employed by
Ethernet is called truncated binary
exponential backoff (TBEB). The source of
information used in this strategy, concerning
the ether congestion, is restricted to a
particular station's experience with respect
to a particular packet.

The author is aware of six different
analytic models of PLWT [3-8], each built for
a different purpose. They may be classified
into two categories: finite and infinite
population models.

The infinite population models [3,4]
circumvent the difficulty in dealing with
complex backoff strategies, such as TBEB, by
assuming that the stations as a whole become
ready at random to access the ether. Under
this assumption, [3] derives the Laplace
transform of delay distribution for an
Assemble packet.  
Let transmission trial count \(a := 0\).  
Defer until ether seems idle.  
Let \(a := a + 1\) and start packet transmission.  
If transmission terminates,  
Then report success to host; end.  
In case collision is detected during transmission,  
Suspend transmission and send jam for collision reenforcement.  
If \(a = 16\),  
give up transmission; report excessive collision error to host; end.  
If \(a \leq 15\),  
backoff \(B\) slots. \(B\) is uniform random integer in \([0, 2^k - 1]\), \(k = \min(a, 10)\).  
\(1\) [slot] = 51.2 \(\mu s\) = 512 [b].  
Try again.

Figure 1 Ethernet data link protocol.

optimal backoff strategy. The result is not directly applicable to the Ethernet's strategy, which is suboptimal [4] is concerned mostly with throughput.

Unlike the infinite population models, the finite population models [5-8] do not allow circumventing considerations on backoff strategies, due to the need to describe the state transition for individual stations. Since Ethernet employs TBEB, a model with the same strategy would suit the present purpose most naturally. [5,6] assume not TBEB but exponentially distributed backoff time. [7,8] can accommodate a version of BEB, which differs from the Ethernet's in that, after a cycle consisting of a nontransmission period and a transmission period, all stations restart transmission trial count from scratch.

The overall structure of the model to be introduced is explained in 2. The model decomposes into two parts: a simple infinite population ether model and a station model attached to it. The ether state model is described in 3. The station model, which receives parameters from the ether model, is found in 4. Remarks in 5, suggesting improvements in the LUT strategy, concludes the paper.

2. OVERALL MODEL STRUCTURE

The model belongs to the class of infinite population models avoiding the backoff strategy formulation, provided that all stations are considered as a whole. However, when a single station is considered, the model must produce delay distribution. This is made possible by decomposition of the system into two submodels, as shown in Figure 2: one concerned with the ether state representing the behavior of collective stations, and the other concerned with the delay distribution describing the behavior of a particular station. The dichotomy is such that an individual cannot affect the rest of the world, while the rest of the world can affect an individual.

Given offered traffic and packet size distribution, the ether model transforms them into the ether's steady state probabilities. Some performance measures, such as throughput, follow from them. The state probabilities are then fed to the station model which, in turn, produces delay distribution. The ether model is by discrete Markov chain; the station model is by convolutions and weighted averages of distributions: both are elementary and straightforward.
3. ETHER MODEL

Time is slotted into $51.2 \mu s$ (= 512 [b]), which Ethernet adopts as the backoff unit time. A transmission is assumed to start only at the beginning of a slot. During a time slot, the ether is in one of three possible states: idle, carrying information, or in collision. The ether is idle when there is no station transmitting signal; carrying information if exactly one is transmitting; in collision if two or more are transmitting. The stations as a whole are assumed to become ready at random to transmit a packet. The Ethernet transceiver holds at most one packet to transmit at a time.

Let symbols be as follow, where "::=" stands for "is defined by":

$N := \text{maximum packet size} = 24 \text{[slots]}.$

$g := \text{offered traffic} \text{[packets/slot]}; 0 \leq g.$

$P[.] := \text{probability}; 0 \leq P[.] \leq 1.$

$r_i := \text{rate the number of size i [slot] packets occupy among all};$

$0 \leq r \leq 1, \Sigma_{i=1}^N r_i = 1.$

$x_0 := P[\text{ether is idle}].$

$x_{ij} := P[\text{ether is transmitting the j-th part of a size i packet}].$

$x_{N+1} := P[\text{ether is in collision}].$

Figure 3 describes the state transition diagram. To avoid cluttering, only size $h$ and $k$ [slot] packets among sizes 1 to $N$ are illustrated in the diagram.

Since $x_{11} = \ldots = x_{ij}$, let

$$x_i := x_{i1} = \ldots = x_{ij}; 1 \leq i \leq N.$$  

The probability conservation is then

$$x_0 + \Sigma_{i=1}^N x_i + x_{N+1} = 1.$$  

The (right to left) state transition is:

$$x_0 = e^{-g} x_0 + \Sigma_{1 \leq j \leq M} e^{-jg} x_j + e^{-g} x_{N+1} ;$$

$$x_i = g e^{-g} r_i x_0 + \Sigma_{1 \leq j \leq M} g e^{-jg} r_i x_j +$$

$$g e^{-g} r_i x_{N+1}, 1 \leq i \leq N ;$$

$$x_{N+1} =$$

$$\{ 1 - (1 + g e^{-g}) x_0 +$$

$$\Sigma_{1 \leq j \leq M} \{ 1 - (1 + jg e^{-jg}) x_j +$$

$$\{ 1 - (1 + g e^{-g}) x_{N+1}.$$

Figure 3  State transition diagram.

4.2A-2-3
The first equation means that the present state is idle, if there is neither new arrival, nor continuation of a previously carried packet. Likewise, the second equation tells that, if there is exactly one station to talk now, then the ether enters the state of carrying a packet. The third captures the complement of the cases considered by the first two. The state transition, as described above, defines a regular Markov chain.

Set auxiliary variables as below, where the sums are over $1 \leq i \leq N$:

\[
\begin{align*}
s &= \text{mean packet size} = \sum i r_i; \\
b &= \sum i \, g e^{-i g} r_i; \\
c &= \sum i \, g e^{-i g} r_i; \\
w &= 1 + s \, g e^{-g} - c.
\end{align*}
\]

The solution to the stationary equation is routinely verified to be

\[
\begin{align*}
x_0 &= e^{-g} \left( 1 + g b - c \right) / w; \\
x_i &= g e^{-g} r_i / w, \quad 1 \leq i \leq N; \\
x_{N+1} &= (1 - e^{-g})(1 - c) - g e^{-g} b / w.
\end{align*}
\]

Throughput follows from the solution.

\[
S := \text{throughput } [\text{slots / slot time}] = P[\text{ether is transmitting information}] = \sum x_i = s \, g e^{-g} / w.
\]

In the special case of a single packet size, using a more popular notation $G := \text{offered traffic } [\text{packets / packet time}] = sg$ and $u := \text{slot size } [\text{packet time}] = 1/s$,

\[
S = G e^{-uG} / (1 + G e^{-uG} - G e^{-G}).
\]

Returning to the case with multiple packet sizes and proceeding as noted in [9],

\[
\begin{align*}
n &= \text{mean number of trials until success} = \text{offered traffic / throughput (both in } [\text{slots / slot time}]\text{)} \\
&= s \, g / S = w e^{g}. \\
p &= \text{P[a single trial turns out a success]} = 1 / n = e^{-g} / w.
\end{align*}
\]

A comparison of three models with different time grains is sketched in Figure 3. The model presented above is the coarsest with grain = slot = 51.2[µs]. The infinite population model in [4] is intermediate with grain = minislot = slot/2 = 25.6[µs]. The simulation model is finest with grain = bit =

![Figure 1 Sample ether state probabilities.](image-url)
0.1[μs]: network diameter 2.5[km] (Ethernet maximum); 1 024 (Ethernet maximum) equally spaced stations; initial state with no packet in system; 5[s] initial data discarded; 100 000 sample packets. Packet size is fixed to 4[kb] because the minislot grain model does not allow mixed sizes. The right half of the graph has no practical meaning because of the excessive collision errors. In the noncritical range, the slot grain model throughput is practically identical to the minislot grain model's throughput. The bit grain model's throughput curve lacks the hump which is characteristic of the analytic PLUT models, looking more as if it were nonpersistent. Simulation results not presented here suggest that throughput's dependence on the number of stations is slight, if there are over a hundred of them.

4. STATION MODEL

Time is again slotted into the backoff unit time, as in the ether model. This time, the model is with respect to a particular station. Let

\[ T := \text{ether acquisition delay [slots]} \]

\[ T = \text{time to start a successful packet transmission after the transceiver has received a packet; } 0 \leq T. \]

\[ \mathcal{A} := \text{accesses [times]} \]

\[ \mathcal{A} = \text{number of trials to start successful packet transmission; } 1 \leq \mathcal{A} \leq \mathcal{A}. \]

\[ K_a := \text{delay due to the } a\text{-th trial [slots]; } 1 \leq a \leq \mathcal{A}. \]

Then,

\[ T = \sum_{1 \leq a \leq \mathcal{A}} T_a; \]

\[ K_a = B_a + D_0 + c \quad 1 \leq a \leq \mathcal{A}, \]

where

\[ B_a := \text{backoff time due to the } (a-1)\text{-th trial, with } B_1 = 0, 1 \leq a \leq 16; \]

\[ c := \text{time required to detect and process collision} \]

\[ = 1 \text{ [slot]}; \]

\[ D_0 := \text{deferrence time when a trial resulted in failure [slots]}; \]

\[ D_1 := \text{deferrence time when a trial resulted in success [slots]}. \]

Note that \( B_a \) is defined as the backoff due to the \( (a-1)\text{-th trial, rather than the } a\text{-th.} \)

This is equivalent to considering that a backoff \( B_a \) is performed before the \( a\text{-th trial, rather than after the } (a-1)\text{-th failure.} \)

The quantities above, represented by capital letters \( (T, \mathcal{A}, K_a, B_a, D_0 \text{ and } D_1) \), are all taken to be random variables.

The distributions for individual random variables are now considered. Assuming the mutual independence of trials, the number of accesses \( \mathcal{A} \) is geometrically distributed in the range \( 1 \leq \mathcal{A} \leq 16: \)

\[ P[1 \leq \mathcal{A} \leq 16] = p \quad q^{\mathcal{A}-1}; q := 1 - p. \]

The remaining probability \( q^{16} \) will be dealt with separately. Since \( B_a, 1 \leq a \leq 16, \) follows the uniform distribution in \( \{0, 1, \ldots, 2\text{min}(a-1,10)-1\}, \) its probability generating function (pgf) is given by

\[ B_a(z) := 1/b(a) + z/b(a) + \ldots + z^{b(a)-1}/b(a) \]

\[ = (1 - z^{b(a)})/(b(a) (1 - z)); \]

\[ b(a) := \min(a,10). \]

Since \( c = 1, \) its pgf is \( C(z) := z. \)

Let

\[ D := \text{deferrence time [slots]}; \]

which is not conditional to the trial result, failure or success. From the ether model,

\[ P[0 = D] = x_0 + \sum_{1 \leq i \leq M} x_i + x_{M+1}, \]

\[ P[1 \leq d = D \leq M] = \sum_{d+1 \leq i \leq M} (i - d) x_i. \]

Consider \( D_0, \) the case in which the trial...
failed. By Bayes's rule,

\[ P[D = 0 | \text{failure}] = \frac{x_M + \sum i \delta^n x_i}{q} \]

\[ P[\text{success} | D = 0] = \frac{\sum_d \delta^{d+1} \sum \delta^n (i-d) x_i (1-e^{-i\theta})}{q}. \]

Similarly, for \( D_1 \),

\[ P[D = 0 | \text{success}] = \frac{r_0}{p} , \]

\[ P[\text{success} | D = 0] = \frac{\sum_d \delta^{d+1} \sum \delta^n (i-d) x_i e^{-i\theta}}{p} . \]

Hence, the pgfs for \( D_0 \) and \( D_1 \) are

\[ D_0^*(z) = \left( x_M + \sum \delta^n x_i \right) + \sum_d \delta^{d+1} \sum \delta^n (i-d) x_i (1-e^{-i\theta}) / q ; \]

\[ D_1^*(z) = \left( r_0 + \sum_d \delta^{d+1} \sum \delta^n (i-d) x_i e^{-i\theta} \right) / p \]

Next, pgf for \( K_a \) will be determined from those for \( A, B, c \) and \( D \). Assuming the necessary independence, the pgfs for \( K_a, 1saA-1, \) and \( K_A \) are, respectively, \( B_a^*(z)D_0^*(z)z \) and \( B_A^*(z)D_1^*(z) \). If \( A \) were equal to a constant \( k \), then the sum \( T_k = \sum_{i=0}^{\infty} K_a \) would have a pgf

\[ T_k^*(z) = \sum_{i=0}^{\infty} K_a \cdot (B_a^*(z)D_0^*(z)z) + (B_a^*(z)D_1^*(z)) \]

\[ = D_1^*(z) (zD_0^*(z))^{k-1} \sum_{i=0}^{\infty} K_a B_a^*(z). \]

Since \( A \) is a random variable, \( T_k^* \)'s distribution is a mixture of \( T_k^* \)'s distributions, with pgf

\[ T^*(z) = \sum_{i=0}^{\infty} K_a B_a^*(z) + q^{a-1} T_0^*(z) + q^{16} z^w \]

\[ = p D_1^*(z) \sum_{i=0}^{\infty} K_a B_a^*(z) + q^{a-1} \sum_{i=0}^{\infty} K_a B_a^*(z) + q^{16} z^w , \]

where the last term is for excessive collision error.

Actual computation is carried out by convolutions and weighted sums of distributions. Three arrays are required to reside in the main memory, the rest being stored in a secondary memory. With a minicomputer, it takes several minutes to compute. An example is illustrated in Figure 5; the simulation condition is as in the previous section, except that the sample size is 12 422. Distributions obtained by simulation tend to have longer tails.

---

**Figure 5** Sample delay comparison.

| packet size | 4 [kb] |
| throughput | 5.1 [Mb/s] |
| offered traffic | 11.7 [Mb/s] |

A: analysis fitted by throughput
S: simulation

0: analysis fitted by offered traffic
5. CONCLUDING REMARKS

Some of the difficulties encountered in modeling Ethernet are due to its following properties: persistence, TBEB, and the existence of the first trial which has to be treated separately from retrials. A nonpersistent protocol is easier to model than a persistent one since there is no need to consider deference. A backoff strategy employing more information than TBEB and closer to an optimal opens a possibility to regard P[some station succeeds] as being load independent. Also, since such a strategy will have a more stable backoff range than TBEB, the backoff time might be treated as a uniformly distributed random number over a fixed interval, given a constant load. Backoff performed before each trial rather than after failure renders unnecessary the distinction between a trial and a retry.

Design of a protocol taking these in consideration not only facilitates modeling but also leads naturally to improved network performance. One such protocol may be found in [10]. Each station continuously observes the channel state and tries to adjust its backoff range so that the sum of idle time observed per collision is close to a preset constant. The protocol was designed for a 32[Mb/s] optical star network with 100 ports, 1[km] diameter, with packet priorities. An analysis method suitable for such protocols, parallel to but simpler than the present paper's, is described in [11]. It is interesting to note that the simulator for the network mentioned above was also simpler and considerably faster than Ethernet's, with the same one bit accuracy.

ACKNOWLEDGEMENTS

The author is indebted to Tomoo Kunikyo, Toshiba Information and Communication System Laboratory, for encouragement and the necessary familiarity with Ethernet. Akira Asai, a Tokyo University student, debugged the formulation, came up with many useful ideas, and did a large portion of the programming. Simulation is by Mutsuoi Fujihara, Toshiba Research and Development Center.

REFERENCES