What is a *local area network*? A local area network is a switching system that

- Employs digital rather than analog transmission
- Transmits bits serially rather than in parallel
- Employs typical clock rates of one to twenty million bits per second
- Is relatively noise free compared to analog voice communication lines: Bit error rates of one bit in one billion are typical
- Switches packets or frames of bits rather than holding transmission bandwidth for the duration of a communication session
- Has a geographic extent of one to ten kilometers at most
- Can have a wide variety of devices attached (e.g., sensors, thermostats, security alarms, process control devices, low speed data terminals, voice, facsimile, computer high speed I/O, video) offering multiple services.

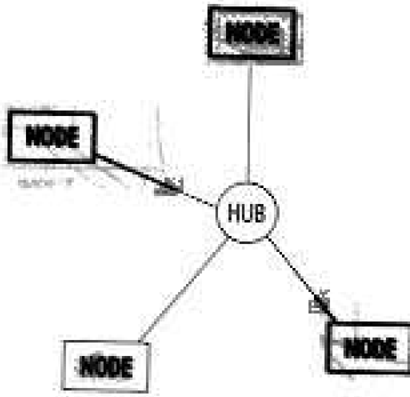
A key design consideration in local area networks is that for typical applications with packet lengths of five hundred to ten thousand bits, at most *one* packet will be in transit at any time, in contrast to long haul terrestrial or space satellite networks where more than one packet (perhaps ten to one hundred packets) may be in transit at any time.

11.1 Traffic Handling Characteristics of Local Area Networks

Topologically, any local area network appears to be a set of stations connected to a common hub or pincushion. This suggests two different traffic handling regimes:

- One regime where the local area network is lightly loaded, the transmission medium has a low utilization, message delays are acceptable, because the workload generated by the stations is too low to cause unacceptable congestion in the network, and
- A second regime where the local area network is heavily loaded, the transmission medium has a high utilization, message delays are unacceptable, because the workload generated by the stations does cause unacceptable network congestion.

The purpose of traffic analysis is to determine the demarcation between the lightly and heavily loaded regions; in any such system it will always be present, the only question is *where*? Here is a different way of thinking about this:



ONE HOP LOCAL AREA NETWORK TOPOLOGY

Figure 11.1. Topology of a Local Area Network

Where would a designer choose to operate a local area network? Since there are more than enough problems getting these systems to work at all, let alone with congestion, many designers would choose to operate in the lightly loaded regime.

11.1.1 *Traffic Analysis Inputs* In order to say *anything* about the traffic handling characteristics of a local area network, we must specify two ingredients, the *workload* (how many devices are attached, how often does each device generate each type of message, how many bits per message), and the *policy* for arbitrating contention for the shared but serially reusable local area network transmission medium.

11.1.2 *Traffic Handling Goals* What are desirable properties of a given policy or access method?

- Packet delay should be acceptable under light load
- The transmission medium should be efficiently utilized under heavy load
- For a fixed workload, packet delay should be *insensitive* to how the workload is generated among the devices since in practice this will *not* in fact be known with any precision.

We will judge traffic handling performance by examining mean packet delay and mean packet throughput rate for a fixed workload. The figure below is an illustrative plot of mean delay versus load: Since we are employing *distributed* control, presumably the delay under light load is *greater* than for central

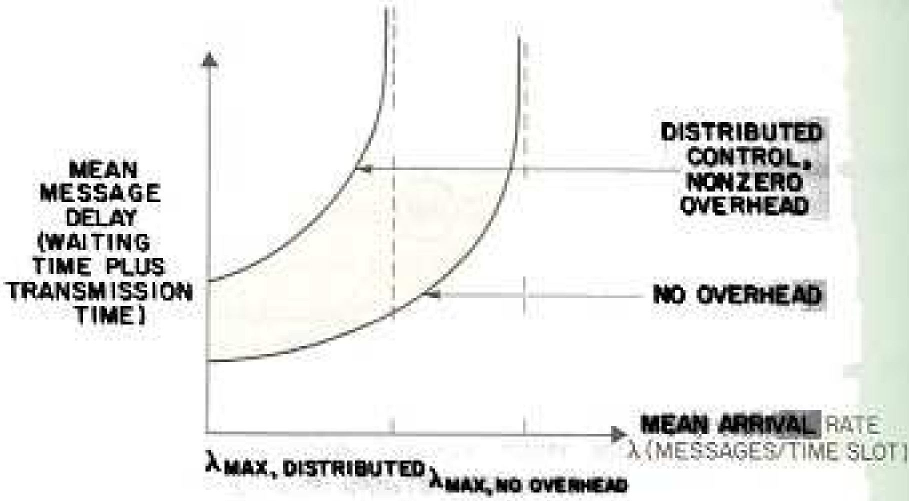


Figure 11.2.Mean Packet Delay vs Mean Packet Arrival Rate

control, and the *maximum* mean throughput rate is *lower*. We want to quantify precisely how much worse these measures can be, for a variety of access methods or scheduling policies for arbitrating local area network access.

We stress that the link level access method is only *one* factor that must be addressed in total communication system performance.

11.1.3 Additional Reading

[1] P.Baran, *On Distributed Communication Networks*, IEEE Transactions on Communications, 12, 1-9 (1964).

11.2 Local Area Network Topology

A local area network might have a variety of topologies. One example is a *star* topology, where all nodes are connected to a central hub. This might be implemented with circuit switching, such as in a private branch exchange, where transmission capacity is dedicated for the duration of a communication session at the start of the session, and if there is none available a new attempt is blocked or rejected. It might be implemented with packet switching, where messages are broken into packets and buffered internally until transmission and switching resources become available. Both these approaches employ *central* control or arbitration of resources.

In contrast to these, a *broadcast* medium or *bus* might be employed, with either *central* or *distributed* control. The transmission attempt of any station is received by all stations. A different type of topology that can employ

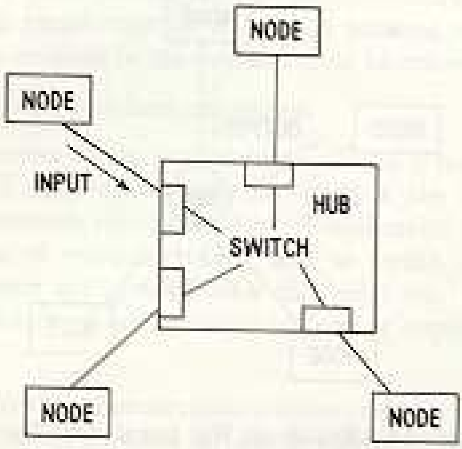


Figure 11.3.Circuit Switching Local Area Network

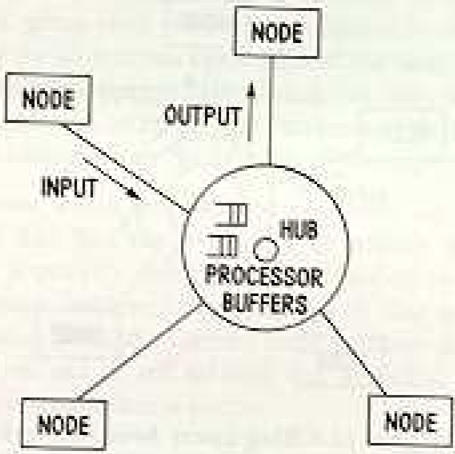


Figure 11.4.Packet Switching Local Area Network

istributed control is a *ring*, where each station receives messages from only ne station and transmits to only one station. This taxonomy by no means xhausts the design alternatives. These topologies are currently the most opular, and we will confine attention to them from this point on.

1.2.1 Additional Reading

[1] J.H.Saltzer, D.D.Clark, K.T.Pogran, *Why a Ring?*, Proceedings Seventh Data Communications Symposium, pp.211-217, 27-29 October 1981, Mexico City, Mexico, ACM 533810, IEEE 81CH1694-9.

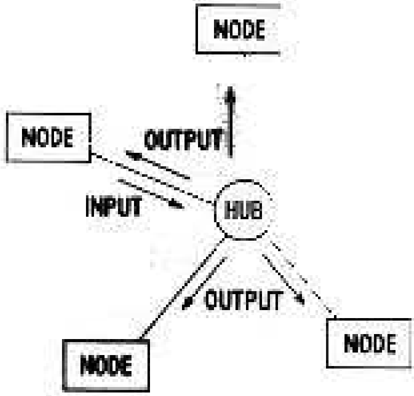


Figure 11.5.Broadcast Bus Local Area Network

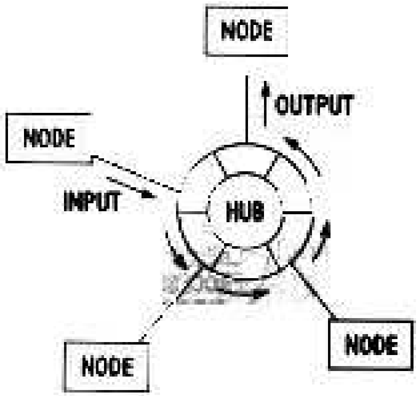


Figure 11.6.Ring Local Area Network

11.3 Bus or Broadcast Medium Model

In this section we focus attention on a bus or broadcast medium local area network, while in subsequent sections other variants are explored. Here, a set of stations are geographically separated, and send and receive messages on a common broadcast medium called a *bus*. What are the states of the bus?

- *Idle*-- No transceiver is active, and all transceivers recognize this within a given time unit
- *Transmit*-- Exactly one transceiver is actively transmitting a message successfully

\P

erferen nsmission atten
nd all messag lved in ist b

p

ok

ke
Colli ons are rare events he: nit

are ignored m th poin Ea tion can
polled the it (from a ce al sta

llision detect

is sensed, ation will att pt to seize the tran nedium

r n atte seize e t mi
dium within giv ime nterval, a olli. aid ccu ind
later time Th nanism for de rmining tr me inte in lve
spreading nsme sion attempts
to maximi

Decision tree a hich
nd like token pa g
chieves this priority arbitrati re y poli a colli
nterrogating (subtrees) of s such

as ack carrier sen llision detection assu es th
me atio ut of stations h packet, while toker

transmission mediu is used fo both ontr d ata

on
red w nt mitt d P unde
loads, while ca en lisio e
Decis

Additional R din

J.E.Donnel een
Prioritized *SM.* *Broad* *N* *work* mput
(1979).

O.Spanic *Modelling* *Loca* *nputer*
Network 79)

11.4 Token Passing Access via a Bus

The figure below shows illustrative operation of a token passing or distributed polling system; stations zero, one, four and seven have messages to transmit.

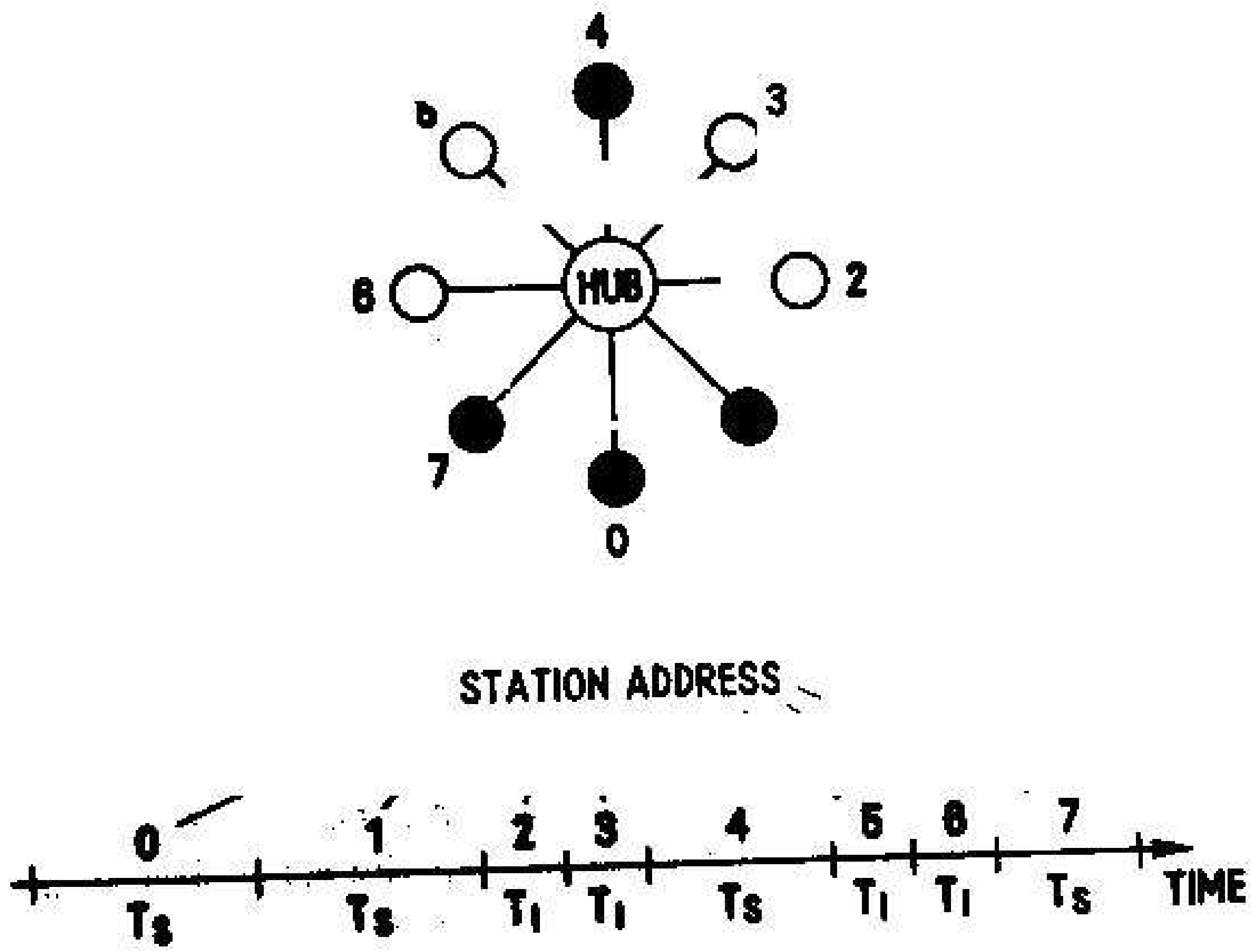


Figure 11.7.Illustrative Token Bus Operation

In the scenario shown in the figure, station zero transmits, then passes control to station one, station one transmits and passes control to station two, and so forth. The mean time to transmit a message is denoted by T_M, and it includes both data bits and framing bits. The mean time to pass the token from one station to another is denoted by T_{TOKEN}.

What are the salient features of distributed polling or token passing transmission medium access? Stations are logically organized in a ring; this allows *separation* or *dedication* of transmission capacity to different types of services, by simply assigning a given number of visits per polling cycle to each service. For example, if a work station offered both 9.6 KBPS data service and 64 KBPS voice service, the voice port could be visited eight times (we have rounded upward from the smallest integer greater than 64/9.6 to get eight) as often as the data port, i.e., we are dedicating eight times the transmission capacity to voice as to data.

Load balancing is possible via two mechanisms: First, token passing can control the number of visits per polling cycle per station (if one station has twice the load of the other stations, visit it twice as often). Second, token passing can control the maximum number of frames transmitted per visit; if one station has a lot of traffic (e.g., a printer, a gateway, or a disk) while the ag.

ee ... the one station with a lot of messages will transmit for a long time if there is no limit on the number of frames transmitted per visit, and all the other stations must wait for it to finish. This can lead to unacceptable delays, but the bus can be very efficiently utilized. If a limit is set on the maximum number of frames that can be transmitted per visit, then each of the terminals or work stations will get access just as under light load, while the one station with lots of messages will be delayed (but it will be delayed anyway since it has so much to transmit, the only question is how much!)

... time per transmission is bounded by a linear function of the number of transceivers (typically this is a constant due to bus propagation and circuitry transients plus a linear term due to the transceiver processing) because control must be passed to each station at least once in a ring or polling cycle.

Under heavy load high throughput is achieved because in this regime the transmission medium will be busy transmitting messages under load, not passing the token from station to station.

... loading on the local area netwo... mean waiting
station ... is ready to transmit a packet or frame equals the time for the
token ... ulate through one half of the tot... number of stations N, on the
averag...

$$T_W = \tfrac{1}{2}\, T_{TOKEN}\, N \quad \textit{one out of N stations active}$$

his immediatel... gives us the maximum mea... ghpu... e fo
always active:

$$\ldots T_{TOKEN} + T_M$$

With ev... n (all N) active, the mean waiting is given by:

$$T_W = NT_{TOKEN} + (N-1)T_M \quad \textit{all N stations active}$$

his in turn ... ives us an upper bound on the mean throughput rate

Suppose the fraction of time the transmission medium is busy transmitting data, the bus utilization, is fixed. As the number of stations is increased so that the amount of data per message per station decreases toward zero, the mean waiting time or delay experienced by any station in transmitting a message will increase above any fixed threshold, because more and more time will be spent passing the control token from one station to another rather than transmitting data. In this sense, token passing is said to be *unstable*.

If we fix the utilization, the fraction of time the bus is busy with data transmission, and increase the number of stations, each station will be less and less likely to have a message to transmit, but we will still pass the token through each station, increasing the total message waiting time. What is the point? Utilization of the local area network transmission capacity may be an inadequate or incomplete measure of system loading: we must also describe how many stations are attached, and how active each of them is, in order to say anything concerning the traffic handling characteristics of such a system.

11.5 Token Passing Mean Value Analysis

This section deals with several limiting cases in order to gain insight into bounds on the mean throughput and delay.

11.5.1 One Message Always at Every Station Assume one message is always ready for transmission at each station. Every station is continually offering work, and the system is never idle.

First, we fix notation. $R_{max}(K)$ denotes the maximum mean throughput rate of messages from source $K, K-1, ..., N$. $T_M(K)$ denotes the mean amount of time required for a message to be transmitted from source $K, K-1, ..., N$ once the serially reusable channel is seized by that source. Source K, $K-1, ..., N$ is visited $V(K)$ times per polling cycle.

First, we assume there is no overhead incurred in token passing, to bound the best possible performance. The fraction of time this serially reusable resource is busy handling requests from source K is simply the mean throughput of that source times the mean service time for that source:

$$U(K) = R_{max}(K) \times T_M(K) \quad K=1,...,N$$

where $U(K)$ is the utilization or fraction of time source K is active. In order to insure that the system can keep up with its work, we demand the fraction of time the serially reusable resource is busy must be less than one:

$$\sum_{K=1}^{N} U(K) = \sum_{K=1}^{N} R_{max}(K) \times T_M(K) \leq 1$$

Now we include polling overhead. During a polling cycle, the link is assumed to be busy either executing work from a source or busy doing overhead work in switching from one source to another. We denote by T_O the total mean time interval that the system is busy executing overhead work during one polling cycle, i.e., the sum of times to move from a source through all the other sources in a cycle and return, with no station transmitting any messages at all. We denote by C_{max} the maximum duration of one complete polling period or cycle. Since every source is assumed to always have a message, we are computing the worst case performance under congestion. Since we assume that the system is busy either doing useful work or doing overhead, the polling cycle must satisfy

$$C_{max} = T_O + \sum_{K=1}^{N} T_M(K) \times V(K)$$

Since we assume that each source always has a message to be sent, the mean throughput rate for source $K, K=1,...,N$ is

$$R_{max}(K) = \frac{V(K)}{C_{max}} = \frac{V(K)}{T_O + \sum_{K=1}^{N} T_M \times V(K)} \quad K=1,...,N$$

11.5.2 Polling Stations that are Idle and Active Assume the system is always busy with polling overhead or message transmission, but the sources may be busy or idle. In the previous section, we calculated the maximum mean throughput rate and the maximum cycle time; here we see

$$R(K) \leqslant R_{max} \quad K=1,...,N \qquad C \leqslant C_{max}$$

Since there is only one reusable message transmission medium, the fraction of time the system spends in the overhead and message states must sum to one.

$$\frac{T_O}{C} + \sum_{K=1}^{N} R(K) \times T_M(K) = 1$$

Solving for C, we get

$$C = \frac{T_O}{1 - \sum_{K=1}^{N} R(K) \times T_M(K)}$$

The duration of a polling cycle is proportional to T_O at all traffic rates: measure this to see if your actual system is performing as expected (doubling the mean overhead time should double the mean cycle time, for a fixed workload). If each source is polled once in a polling cycle, the reciprocal of the polling cycle is an upper bound on the mean throughput rate for each source. More generally,

$$R(J) \leqslant \frac{V(J)}{C} \quad J=1,...,N$$

$$R(J) \leqslant V(J) \times \frac{- \sum_{K=1}^{N} R(K) \times T_M(K)}{T_O}$$

The figure below plots an illustrative feasible region of mean throughput rates for the special case of two sources. This is a convex set, with the convex hull given by the static priority policies.

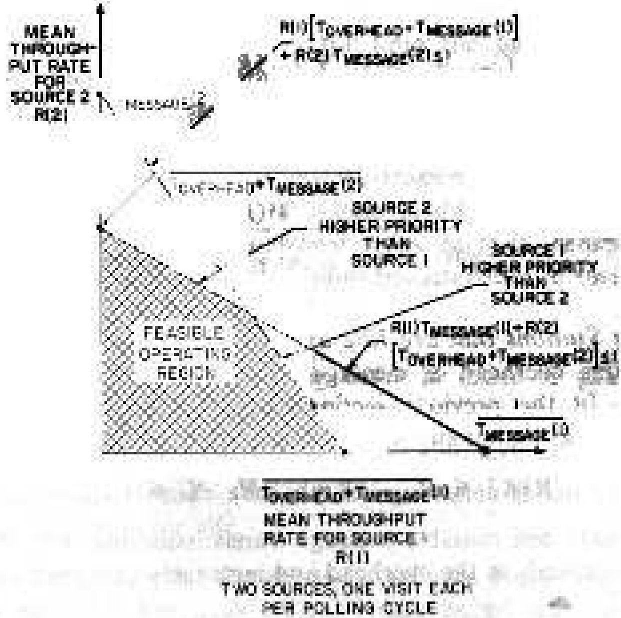


Figure 11.8. Illustrative Mean Throughput Feasible Region

11.5.3 All Sources Generate Identical Workload If the sources have identical traffic characteristics, so that

$$R(K) = R \quad T_M(K) = T_M \rightarrow$$

$$R \leq \frac{1 - N \times R \times T_M}{T_O}$$

$$R \leq \frac{1}{T_O + N \times T_M}$$

Now let us examine the mean delay for each source here. We observe that each source is busy (either queued or transmitting data) for a mean time interval denoted T_D, or is idle for a mean time interval T_I. The mean throughput rate for each source, by definition, is simply the reciprocal of the sum of these two time intervals:

$$R = \frac{1}{T_D + T_I}$$

If we rearrange this, we see that

$$T_D = \frac{1}{R} - T_I$$

Using the earlier upper bound on throughput for identical sources,

$$T_D \geq T_O + (N \times T_M) - T_I$$

In practice, T_O may be a constant dependent on the round trip propagation time plus a term proportional to the number of source visits per cycle

$$T_D \geqslant C_0 + [N \times (C_1 + T_M)] - T_I$$

The figure below shows an illustrative feasible region of mean delay for a given source.

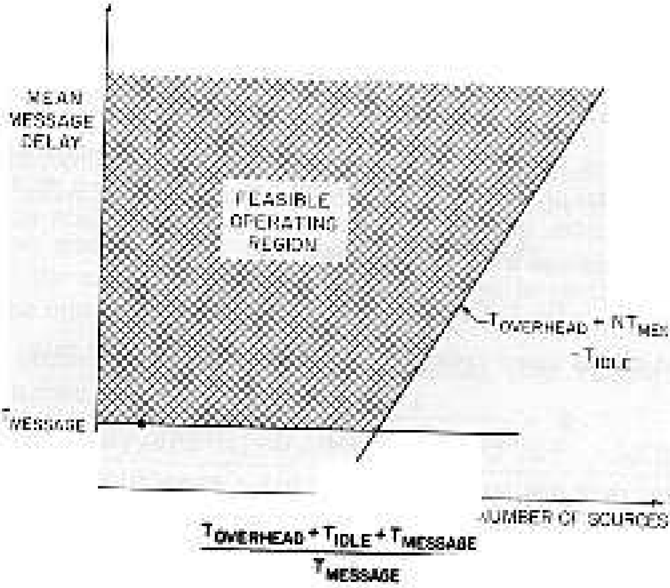


Figure 11.9.Mean Delay vs Number of Stations

There are two regions evident:

- Light loading--the serially reusable resource is lightly loaded, so the mean delay is simply the time required to transmit data, with virtually no queueing; the different sources are the bottleneck in this regime, not the serially reusable resource! The different sources cannot generate enough message to congest the link.
- Heavy loading--the serially reusable resource is heavily loaded, so the mean delay, queueing plus data transmission, increases linearly with each additional source; the serially reusable resource is the bottleneck

If we poll a given source more frequently, the throughput for that source would increase, and the slope of $T_D(K)$ would decrease in the heavily loaded region. However, this might be accomplished at the expense of increasing the polling period, thereby reducing the upper bound on throughput for other sources.

A key concept is that of *load balancing*. The idea here is to configure the system so that the fraction of time spent per visit per station during a polling cycle is roughly equal or balanced. For example, if there are three stations, and during heavy loading one station typically has two messages per cycle while the other stations have one message per cycle, then we could allow each station to transmit a maximum of one message per cycle, and visit the heavily loaded station twice per cycle. Put differently, if we put no limit on the maximum amount of messages that a station can transmit per visit or poll, then any station can drive the system to complete utilization, but the delay might be unacceptable for some stations. If we put a limit on the maximum amount that any station can transmit per poll, then no station can drive the link to complete utilization, but the delay might be made acceptable.

11.5.4 Finite Source Mean Throughput and Mean Delay Asymptotes What if we have one terminal or source on the system, i.e., N=1? Then we see that the source is idle or active, and the active time is the time to gain access to the channel plus the message transmission time.

$$T_O + T_M + T_I = C \quad \textit{one terminal}$$

The reciprocal of this mean cycle time is the mean throughput rate

$$R = \frac{1}{T_O + T_M + T_I} \quad \textit{one terminal}$$

If N is greater than one, and each source has a message to transmit on each visit, then

$$N \times T_O + N \times T_M + T_I = C \quad \textit{N terminals}$$

$$\rightarrow R = \frac{N}{N \times T_O + N \times T_M + T_I} \quad \textit{N terminals}$$

which for large N and fixed T_I will approach that of a single source!

A different type of insight is gained if we start with our fundamental identity

$$\frac{U}{T_M} = \frac{N}{T_I + T_D}$$

We now fix the quantity $N/T_I \equiv \lambda$ at a constant but allow the number of stations and the mean idle time per station to approach infinity together:

$$\lim_{N \to \infty} \frac{U}{T_M} = \frac{U_{max}}{T_M} \quad \lambda = \textit{constant} = \frac{N}{T_I}$$

where U_{max} is the maximum utilization of the serially reusable resource. All mean throughput and mean delay studies are devoted to calculating U_{max} for different workload statistics, and access policies.

EXERCISE: Can you calculate a lower bound on mean throughput rate and a corresponding upper bound on mean delay?

11.6 Exhaustive Polling of Two Queues with Switching Overhead

Here is a case study to illustrate the complexity of analyzing the traffic handling characteristics of exhaustive polling of two stations, which is a case that can be completely handled. The model ingredients are:

- The message arrival statistics--Each station has simple Poisson arrival statistics with mean arrival rate λ_K to station K=1,2.
- The message length statistics--The sequence of message lengths for either station are independent identically distributed random variables, denoted by B_K bits for station K=1,2.
- The scheduling policy--Each station is polled to exhaustion upon visiting a station; messages are queued in order of arrival for transmission.

We are also given the link transmission speed, BPS, measured in bits per second, and the total time to switch from one queue to another and back again given that no message is present at either queue, denoted $T_{overhead}$.

The mean queueing time at queue one, including both waiting and message transmission time, is

$$E(T_{Q1}) = \frac{\lambda_2 E(B_2^2/BPS^2)(1-U_1)^2+\lambda_1 E(B_1^2/BPS^2)U_2^2}{2(1-U_1)(1-U_1-U_2)(1-U_1-U_2+2U_1U_2)} + \frac{\lambda E(B_1^2/BPS^2)}{2(1-U_1)}$$

$$+ \frac{(1-U_1)T_{overhead}}{2(1-U_1-U_2)} \quad U_K = \lambda_K E(B_K)/BPS \quad K=1,2$$

By interchanging indices one and two, a similar expression follows for the mean queueing time at queue two.

In order to gain engineering insight into the behavior of this system, we vary the following parameters:

- The fraction of total arrivals to the system that arrive at queue one $F(1)$
- The mean number of bits per message at queue one and queue two
- Fluctuations in the message length about its mean value for queues one and two, measured via the *squared coefficient of variation* which is the ratio of the variance divided by the mean squared, so it measures fluctuations (standard deviation about the mean) in units of mean message length; zero squared coefficient of variation is no variation at all, squared coefficient of variation of one is the exponential distribution, and so forth

Our goal is to plot the mean delay of messages from queue one and queue two as a function of the total utilization of the transmission medium. Not surprisingly, these figures show that the station with the greater utilization, i.e., the station that is busy the greater fraction of time, will monopolize the

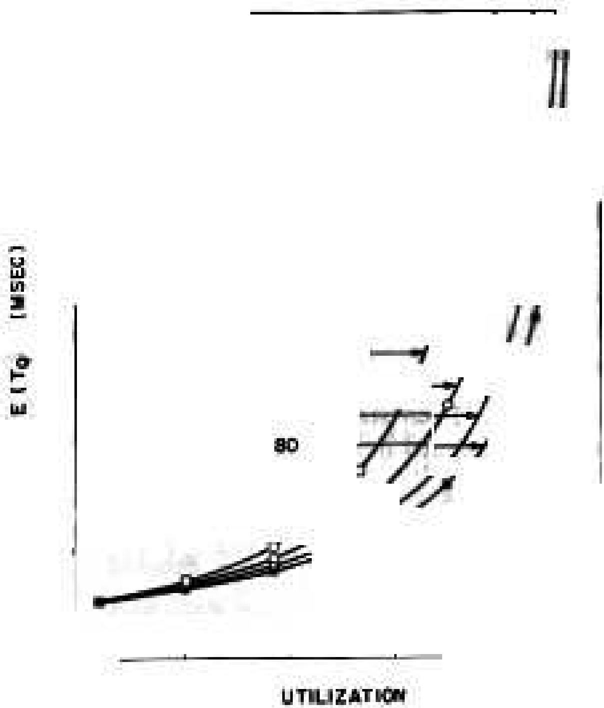


Figure 11.10. $E(T_{Q1})$ **vs Total Bus Utilization**
10 μsec Switching Overhead per Poll, Eight Bit Header/Packet
1 MBPS Transmission Speed, 500 Bits/Packet for Queue 1 and 2
Squared Coefficient of Variation Packet Length: Queue 1 = 1, Queue 2 = 2

transmission medium, and the delays (icountered by message with lighter utilization will be higher t an if the other station re not present. This is presented to show that even fc something supposedly *simple* as twc queues, we should measure

:ket length statis

1.6 dditional Readir

.S.Syk *Analysi 4lternating Prior. Queueing* Research, 18 (6 182-1192

M isenberg. *Twc Queues with Change ,er Time* Ope itions R(arch, 19 -401 (19

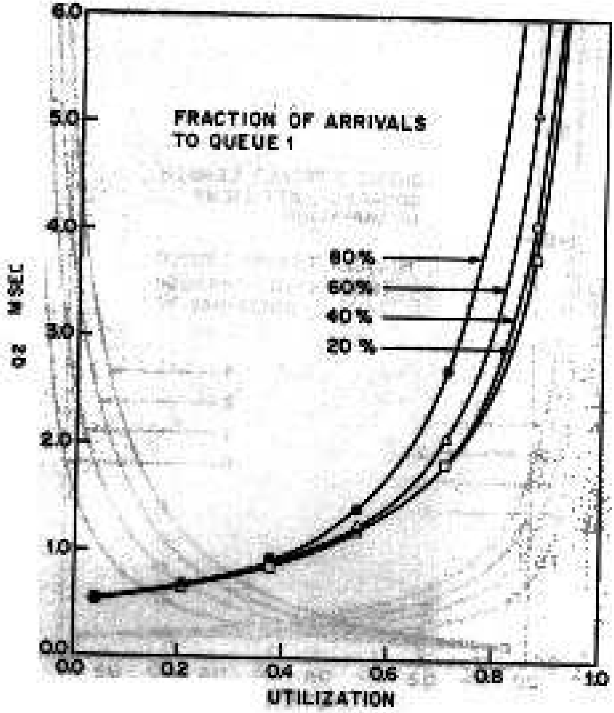


Figure 11.11. $E(T_{Q2})$ **vs Total Bus Utilization**
10 μsec Switching Overhead per Poll, Eight Bit Header/Packet
1 MBPS Transmission Speed, 500 Bits/Packet for Queue 1 and 2
Squared Coefficient of Variation Packet Length: Queue 1 = 1, Queue 2 = 2

11.6.2 Model Here is one model for analyzing performance of a token passing or polling system:

- N stations each generate messages according to simple Poisson arrival statistics, with mean message interarrival time $1/\lambda$ (each station generates messages with the *same* arrival statistics)
- The messages form a sequence of independent identically distributed random variables drawn from an arbitrary distribution with message transmission time denoted by T_M (each station generates messages with the *same* message length statistics)
- The stations are visited in a cyclic manner; after all messages are removed from the buffer at one station, i.e., the service is exhaustive, a time interval of duration T_O passes doing overhead work before reaching the next station; all messages are removed from that next station, including both messages present at the arrival to that station and messages that arrive during the transmission of the initial workload; finally, the next overhead time interval is entered

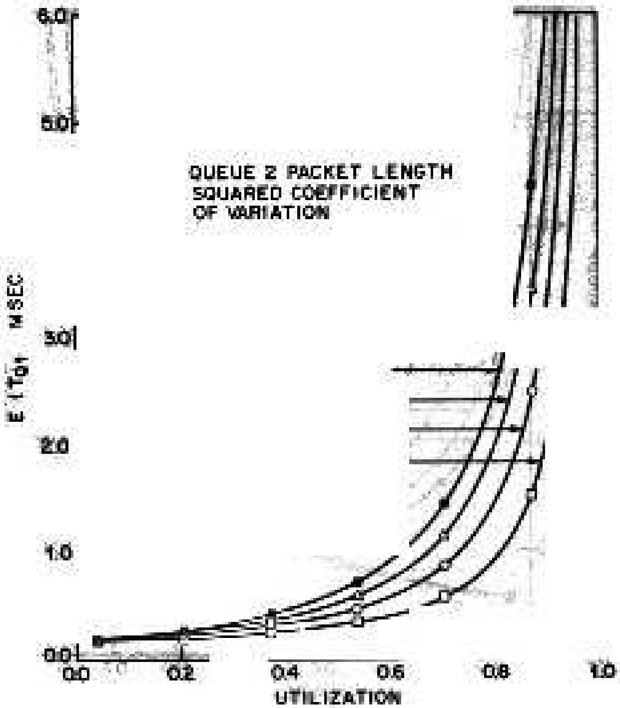


Figure 11.12. $E(T_{Q1})$ vs Total Bus Utilization
10 μsec Switching Overhead per Poll, Eight Bit Header/Packet
3 MBPS Transmission Speed, 500 Bits/Packet for Queue 1 and 2
20% Total Arrivals to Queue 1; Queue 1 Packet Length Sq Coef Var=1

Granted these assumptions, it can be shown that the mean waiting time (the time a message is delayed due to waiting for the token plus waiting for all earlier arrivals to be transmitted) is given by

$$E(T_W) = \frac{N\lambda E(T_M{}^2)}{2(1 - N\lambda E(T_M))} + \frac{1}{2}E(T_O)\frac{1 - \lambda E(T_M)}{1 - N\lambda E(T_M)}$$

The mean waiting with zero overhead, with messages transmitted in order of arrival, gives a *lower* bound on the mean waiting time with polling:

$$E_{polling}(T_W) = E_{order\ of\ arrival}(T_W) + E_{overhead}(T_W) \geq E_{order\ of\ arrival}(T_W)$$

$$E_{order\ of\ arrival}(T_W) = \frac{N\lambda E(T_M{}^2)}{2(1 - N\lambda E(T_M))}$$

$$E_{overhead}(T_W) = \frac{1}{2}E(T_O)\ \frac{1 - \lambda(T_M)}{1 - N\lambda E(T_M)}$$

The first term depends on both the first and second moments of the message length statistics, as claimed. The second term in the mean waiting time expression using polling is due to irregularities or fluctuations in the cycle

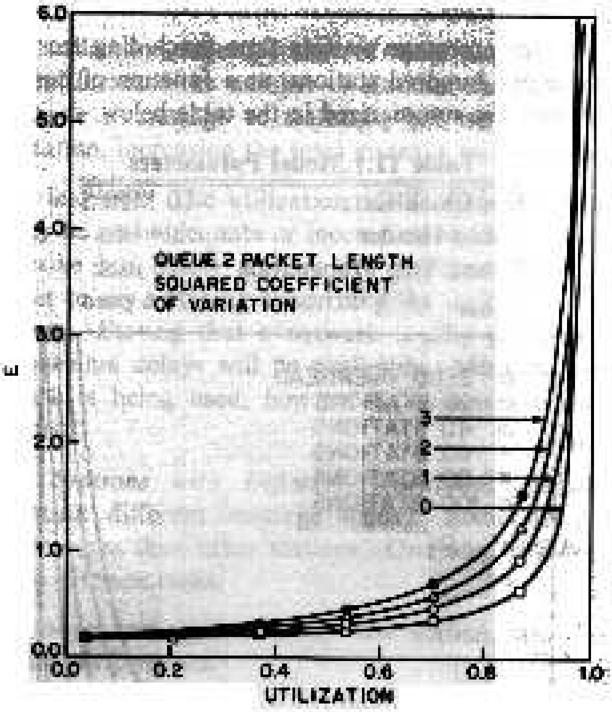


Figure 11.13. $E(T_{Q2})$ vs Total Bus Utilization
10 μsec Switching Overhead per Poll, Eight Bit Header/Packet
3 MBPS Transmission Speed, 500 Bits/Packet for Queue 1 and 2
20% Total Arrivals to Queue 2; Queue 1 Packet Length Sq Coef Var = 1

times

The mean delay, queueing plus transmission, is given by

$$E(T_D) = E(T_W) + E(T_M)$$

This analysis is useful for developing a great deal of insight into the behavior of communication systems.

11.6.3 An Example: N Stations Generating Identical Message Traffic A communication system consists of a single serially reusable resource, a serial bus, shared by N stations. The bus utilization or offered load is fixed, measured in the fraction of time that the transmission medium is busy transmitting data, and excluding the fraction of time the transmission medium is used for token passing control. As the number of stations is increased so that the amount of data per message per station decreases toward zero, then the mean waiting time or delay experienced by any station in transmitting a message will increase above any fixed threshold, because more and more time will be spent passing the control token from one station to another rather than transmitting data. In this sense, token passing is said to be *unstable*. The figure below plots mean message waiting time (excluding message transmission time) for twenty to one hundred stations, as a function of bus utilization. The remaining parameters are summarized in the table below.

Table 11.1.Model Parameters

Bus Clock Rate	10 MBPS
Token Header	96 Bits
Round Trip Propagation	20 μsec
T_{token}	29.6 μsec

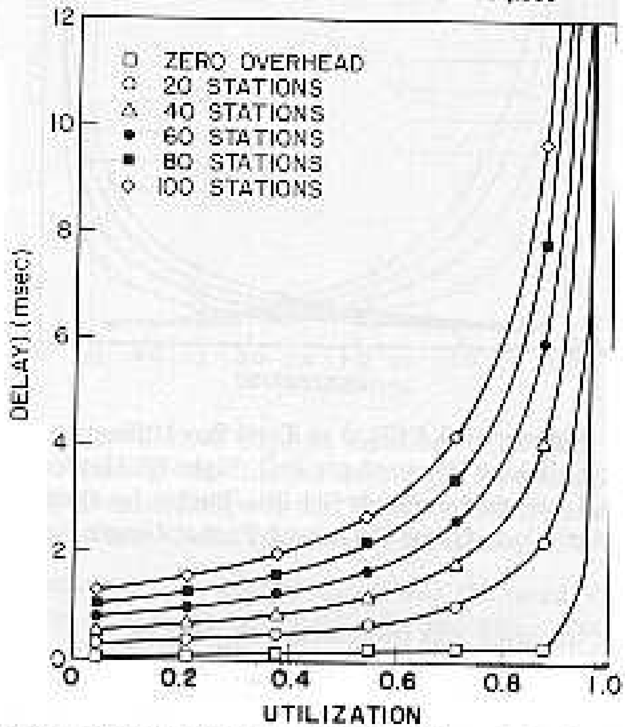


Figure 11.14.Mean Message Waiting Time vs Bus Utilization

Under light loading on the bus, a station will have to wait for the token to circulate through one half of the total number of stations N before it will be allowed to transmit:

$$T_W = \tfrac{1}{2}\, T_{token}\, N \quad \textit{light loading}$$

Each message is assumed to be one thousand bits long, and hence requires one hundred microseconds to be transmitted, T_M. This determines the maximum mean throughput rate:

$$\textit{maximum mean throughput rate} \quad \frac{\quad}{T_{token} + T_M} \quad \textit{messages/sec}$$

If we fix the utilization, the fraction of time the bus is busy with data transmission, and increase the number of stations, each station will be less and less likely to have a message to transmit, but we will still pass the token through each station, increasing the total message waiting time.

What have we learned? The utilization of the serially reusable resource, the link or bus, may be an inadequate or incomplete measure of system loading: we must also describe how many stations are attached, and how active each of them is, in order to say anything concerning the traffic handling characteristics of such a system. Stating that a network is fifty per cent loaded, does *not* imply than congestion delays will be negligible. More information concerning *how* the network is being used, how *many* stations are doing what sorts of things are required.

11.6.4 Polling Stations with Different Workloads What about different stations generating different message loads? Some stations generate and receive more messages than other stations. One way to handle this is to think of bounding two extreme cases:

- All the messages are generated by one station, which will be the *best* possible delay performance
- All the messages are generated equally by all stations, which will be the *worst* possible delay performance

To do the best case, we assume an intervisit model where the token is passed to $(N-1)$ other stations and then to the station with all the work.

To handle the worst case, we use the formula given in the earlier part of this section.

Combining these two analyses, we plot the mean message queueing time versus bus utilization below using ten megabit transmission speed, and typical protocol header bits, and bus propagation time. Always plot these graphs to see how important imbalance is *first* before jumping to conclusions! For twenty stations, for practical purposes the two bounds are identical; for one hundred stations, the two bounds *can* be far apart. For twenty stations, no more analysis may be needed; for one hundred, more data should be gathered on message traffic.

11.6.5 Terminal Polling vs Host Computer Polling Measurements are carried out on a communication system. First, the frequency of arriving messages per terminal is measured:

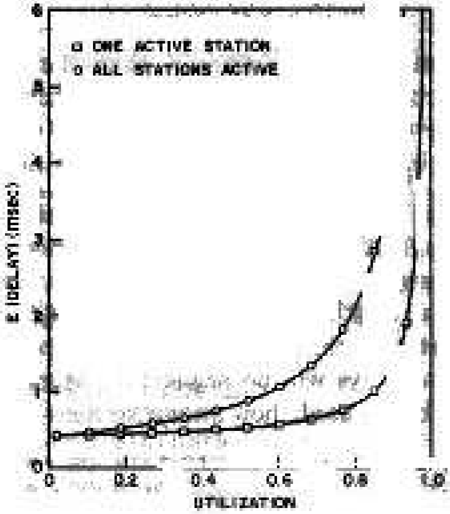


Figure 11.15.A.Mean Message Queueing Time vs Data Link Utilization, 20 Stations

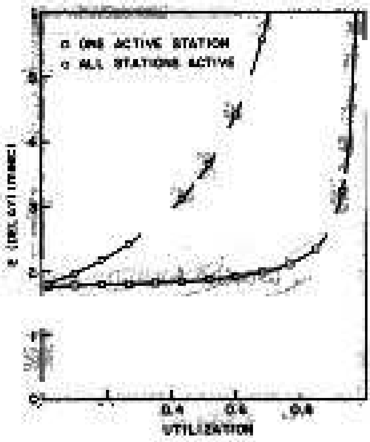


Figure 11.15.B.Mean Message Queueing Time vs Data Link Utilization, 100 Stations

Table 11.2.Terminal Arrival Statistics

Messages per Hour	*Interarrival Time(sec)*	*Messages per Hour*	*Interarrival Time(sec)*
10	360	30	120
15	240	35	103
20	180	40	90
25	144	45	80

The terminal transmission statistics are given by the following distribution:

Table 11.3. Terminal to Computer Transmission Statistics

Percentage	*Number of Eight Bit Characters*
40%	80 characters
20%	81-320 characters--mid range 200 characters
20%	321-700 characters--mid range 510 characters
20%	701-1500 characters--1100 characters

The distribution of the number of characters received by each terminal is given in the table below:

Table 11.4. Computer to Terminal Transmission Statistics

Percentage	*Number of Eight Bit Characters*
20%	80-240 characters--mid range 160 characters
20%	241-480 characters--mid range 360 characters
20%	481-720 characters--mid range 600 characters
20%	721-1440 characters--mid range 1080 characters
20%	1441-1920 characters--mid range 1680 characters

We are interested in the mean number of characters transmitted (both ways) per polling visit to an active station, and its variance; this is simply the sum of the means and variances, respectively, of the total number of characters transmitted both ways per polling visit.

$$C_S^2 = \frac{\textit{variance of number of transmitted characters}}{[\textit{mean number of transmitted characters}]^2}$$

Note that the message distribution may be adequately modeled by an Erlang 3 distribution (why?). Generate a quantile-quantile plot to check this!

For the numbers of interest here, using the above distributions, we see

$$\textit{mean number of transmitted characters} = 1176 \textit{ characters}$$

$$\textit{variance of number of transmitted characters} = 447{,}968 \textit{ characters}^2$$

$$\textit{squared coefficient of variation} = 0.3272$$

Thus, the fluctuations are not nearly as great as an exponential distribution would generate, based on this criterion.

We now combine all this information to determine the desired number of terminals.

- Two configurations are considered, one with host computers alone, one with terminals alone
- The mean interarrival time of messages from terminals (30 seconds) or from host computers (30 msec)

- The transmission speed of the link (one, three and ten megabits per second)
- The number of characters required to interrogate a given terminal to determine if it has a message or not, called a flag (eight to twenty four bits)
- The one way propagation time of a bit (one to ten microseconds)
- Message length statistics: mean value of one thousand to two thousand bits per message, with a squared coefficient of variation of zero or one. The one way propagation time was fixed at five microseconds.

he tables below summarize the results of various calculations for such a ystem:

Table 11.5.Maximum Number of Sources with Mean Response 10 msec or Less

One Megabit/Sec Line Speed--Mean Message Size=1000 Bits
Squared Coefficient of Variation for Message Length=0

Interarrival Time	*Number of Bits of Overhead/Source*			
	0 bits	*8 bits*	*16 bits*	*24 bits*
30 sec	29,995	2,093	1,085	732
30 msec	30	30	30	29

Table 11.6.Maximum Number of Sources with Mean Response 10 msec or Less

Three Megabit/Sec Line Speed--Mean Message Size=1000 bits
Squared Coefficient of Variation for Message Length=0

Interarrival Time	*Number of Bits of Overhead/Source*			
	0 bits	*8 bits*	*16 bits*	*24 bits*
30 sec	89,973	6,708	3,484	2,353
30 msec	90	89	88	87

Table 11.7.Maximum Number of Sources with Mean Response 10 msec or Less

Ten Megabit/Sec Line Speed--Mean Message Size=1000 bits
Squared Coefficient of Variation for Message Length=0

Interarrival Time	*Number of Bits of Overhead/Source*			
	0 bits	*8 bits*	*16 bits*	*24 bits*
30 sec	299,910	22,858	11,882	8,028
30 msec	300	297	293	290

In addition, the mean message length was varied to two thousand bits per message, which halved all the above numbers, the squared coefficient of variation was varied to one, which changed the above numbers by at most hundredths of a percent, the threshold was varied to twenty milliseconds with virtually no change on the above numbers, and the one way propagation time was varied from zero to thirty microseconds with virtually no impact.

he numbers chosen here show that

- The terminal interarrival time per message, line speed, and mean message size are significant in determining traffic handling characteristics
- The overhead per source visit, the threshold chosen, and the one way propagation time have less of an impact on system behavior

Again, we stress that a variety of other factors, such as the hardware and software design for the communications interface, can and must be considered in assessing this approach for a given application.

11.6.6 Additional Reading

[1] R.B.Cooper and G.Murray, *Queues Served in Cyclic Order*, Bell System Technical Journal, **48** (3), 675-689 (1969).

[2] R.B.Cooper, *Queues Served in Cyclic Order: Waiting Times*, Bell System Technical Journal, **49** (3), 399-413 (1970).

[3] A.G.Konheim, B.Meister, *Waiting Lines and Times in a System with Polling*, J.A.C.M., **21** (3), 470-490 (1974).

[4] P.J.Kuehn, *Multiqueue Systems with Nonexhaustive Cyclic Service*, Bell System Technical Journal, **58** (4), 671-698 (1979).

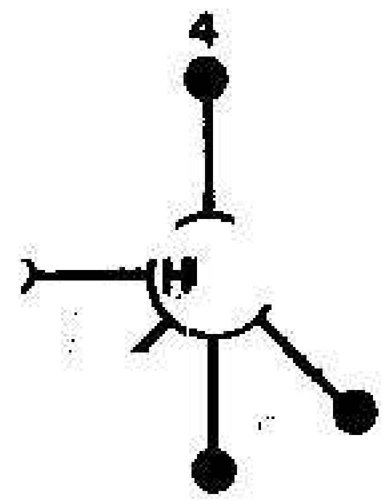
11.7 Carrier Sense Collision Detection

The figure below shows illustrative operation of a carrier sense collision detection system; stations zero, one, four and seven have messages to transmit, just as in the token passing polling case. In the scenario sh. wn there, two collisions occur, and stations seven and four transmit earlier than in the token passing access. In the illustrative scenario shown, stations zero and one attempt to seize the channel simultaneously within a collision window; both schedule retransmissions for a later point in time, and so forth until all messages are transmitted.

Almost immediate media access is possible under light traffic. Put differently, the waiting time is zero under light load, and hence

T_W 0 *one out of N active* →*mean throughput rate* T_M

If a station can exhaustively transmit, then the utilization of the transmission medium can be driven to one hundred per cent. However, this can lead to unacceptable delays as some stations are locked out by other stations. Traffic can be controlled by limiting the maximum number of frames transmitted per local area network access by each station; this allows stations that rarely generate messages a chance to access the local area network, while delaying those stations with a lot of message traffic. On the other hand, the utilization of the transmission medium cannot be driven to one hundred per cent. Time required to resolve conflicts is unbounded because *no* station is successful in



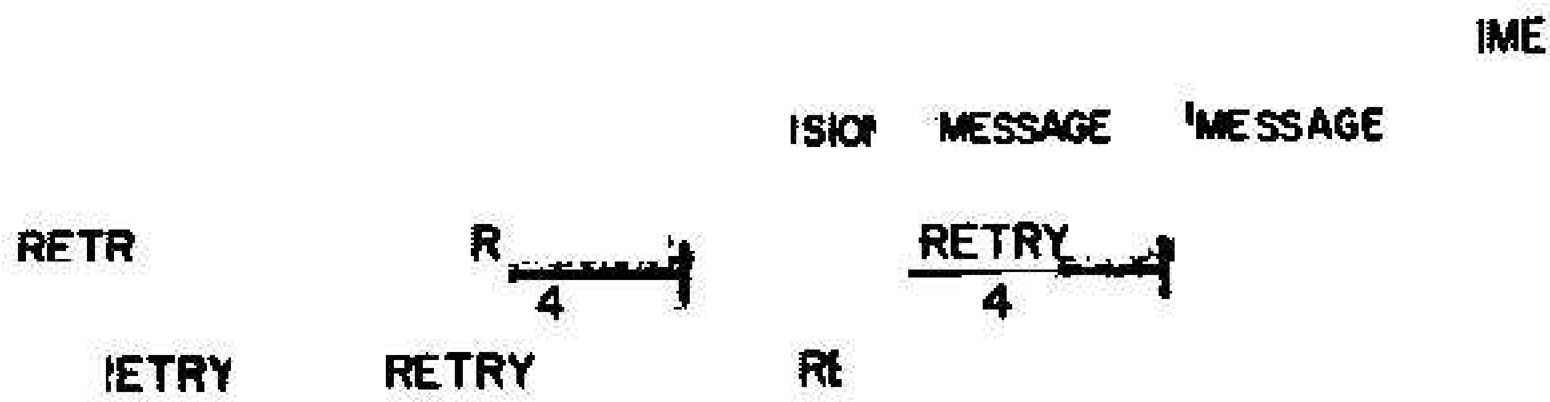


Figure 1.16. Illustrative Operation of CSCD

transmitting its message when a collision occurs (positive feedback!). If the total offered load (bits per unit time) is fixed but the number of stations is increased, more and more collisions will take place to transmit one message, hence the mean delay and message waiting time will increase above any threshold. Utilization is not a complete measure of the loading on a carrier sense collision detection local area network, and more information must be given about the number of stations and what each station is doing in order to say *anything* concerning delay. The mean waiting time to transmit a message grows faster than any linear function of the number of stations attached to the local area network.

A fundamental parameter in a carrier sense collision detection local area network is the *slot time* (denoted here by T_{slot}) which roughly speaking is the worst case time required for a signal to propagate from one end of the network to another, plus account for circuitry transients. A variety of theoretical analyses for carrier sense collision detection access for a local area network have suggested that if N identical stations are connected to the bus, with each station always idle, the maximum mean throughput rate of this system is

$$\textit{throughput rate} \leqslant \quad T_M + (2e-1)T_{slot}$$

In words, there will be roughly 2e=5.43.. collisions for every successful message transmission. This has *not* been validated by controlled experimentation on actual networks. A great deal of evidence (both analytical and simulation) suggest this is in fact a very reasonable first cut sizing of maximum mean throughput rate.

What about mixing voice and data with such an access policy? Again, there are no publicly available and independently reproducible results to date, but there is a body of evidence that suggests the following scenario is quite plausible. We will assume voice will require 64 KBPS of transmission capacity for the duration of a voice telephone call, say one hundred seconds. With a 10 MBPS transmission speed, we can have at most the following number of simultaneous voice telephone calls:

$$\textit{maximum number of simultaneous voice telephone calls} = \frac{10\ MBPS}{64\ KBPS} \approx 156$$

For virtually any scenario proposed to date involving digital voice and low speed data transmission, the total number of bits transmitted will be predominantly voice: check this yourself. People typically make three to six telephone calls per peak business work hour, lasting two to three minutes each on the average, which generates a lot of voice bits relative to the amount of data bits each person might generate.

If we have much less than this maximum number of simultaneous voice telephone calls in progress, then it may be possible for the data messages to be interleaved with voice packets. If we have much more than this number of simultaneous voice telephone calls in progress, because we have many many more voice packets than data packets the voice packets will be more much more likely to be successfully transmitted (this is the nature of the access method), voice packets will swamp the transmission capacity, and data packets will be locked out from using the local area network.

Suppose we have much less than the maximum number of simultaneous voice telephone calls in progress, say 100 simultaneous voice telephone calls during a peak business work hour. Rarely, say one per cent of the time, there will be a surge or fluctuation about this mean value, and we will have *all* the available transmission capacity occupied with voice packets. There are thirty six hundred seconds in one hour, so one per cent of the time, or thirty six seconds out of the hour, the local area network will be completely busy with voice: as far as data is concerned, the local area network will have failed, i.e., there is no transmission capacity for data. Furthermore, the duration of these surges of voice will be ten seconds here, five seconds there, fairly unpredictable, throughout a business hour. Remember there are higher level protocols for flow control with timeouts: these additional control mechanisms were not designed for handling voice and data, only data, and as far as data is concerned, the local area network has failed at this point. The question is not *will* this occur but rather *how often* will this occur: in order to answer that, we clearly need to be more specific about the voice services and data services, but the phenomenon we have just described must be present: the time scale for voice is tens of seconds, while the time scale for data is tens of milliseconds, and hence a short fluctuation in voice load looks like it lasts for a very very long time in the world of data.

At the present time, our understanding of carrier sense collision detection access is relatively incomplete compared with that for token passing access: we understand how one station loads the system, and how an infinite number of stations load the system, but we have no idea of any intermediate (such as *two* stations) case, either in terms of simulations or analysis or data (best of all), for one service or a mix of services.

11.8 Carrier Sense Access Loss Model Analysis

We will analyze two modes of operation, one without any synchronization between stations and one with a clock synchronizing station message transmission attempts. The basic period of the clock is called a *time slot,* and hence the two analyses are for *unslotted* and *slotted* operation, respectively.

11.8.1 Unslotted Operation with Loss All attempts are assumed to obey simple Poisson arrival statistics with mean arrival rate of messages λ. Since there is no synchronization between source arrivals, i.e., they occur randomly in time, this mode of operation is called *unslotted.* The message transmission types are independent identically distributed random variables drawn from an arbitrary distribution denoted by $G_{T_M}(X) = PROB[T_M \leq X]$ with associated Laplace Stieltjes transform $\hat{G}_{T_M}(z)$. If a source has been listening to the channel for at least T_{OV} time units, and the channel is busy with another message over that time, then the attempt from that source is rejected or lost, and presumably will retry later. The moment generating function for the random variable for the time interval between successful message transmission completion epochs, denoted T_S, is given by, from these definitions

$$E[exp(-zT_S)] =$$

$$\frac{\lambda}{\lambda + z} \quad {}^{T_o}\hat{G}_{T_M}(z) + \int_0^{T_o} \lambda e^{-\lambda X} dXe \quad {}^{X}\hat{G}_{\Delta}(z)E(exp \quad zT_S))$$

where Δ is the smallest value of t such that the backward recurrence time from an arrival epoch of a Poisson process with rate λ is T_O. For example, the mean duration of time between successful message transmissions is

$$E(T_S \quad E(T_M) \quad T_O + \frac{e^{2T_O\lambda}}{[illegible]}$$

The mean utilization of the channel is denoted by U, where

$$U = \frac{E(T_M)}{E(T_S)}$$

This reaches its maximum value when $2\lambda T_O =$

$$\max U = \frac{E(T_M)}{E(T_M) + T_O(2e-1)}$$

The interpretation of this is that for every successful message transmission, under maximum link utilization, there will be $2e-1 \approx 5.39$ collisions for every successful transmission. By way of contrast, in a space satellite link, we would choose $E(T_M)$ equal to T_O, i.e., each message will take as long to transmit as it takes to propagate from the transmitter to the receiver, so

$$\max U = \frac{1}{2e} \quad E(T_M) = T_O$$

As an example, suppose that a number of asynchronous terminals can all transmit over a shared 9.6 KBPS channel to a central computer, and if the terminal packet is received without error the central computer broadcasts back to all the terminals over a separate 9.6 KBPS channel an acknowledgement. If no acknowledgement is received, the terminal will try again. An operator at a terminal can type one to two key strokes per second, or ten to twenty bits per second; we will split the difference and call it fifteen bits per second. How many terminals can the link support? If the link is a radio link, with a slot time of 50 μ seconds, then

$$\max U = \frac{15\ bits/9600\ bps}{15\ bits/9600\ bps + 50\ \mu sec\,(2e-1)} \approx 85.3\%$$

With perfect scheduling, each terminal demands 15 bps of bandwidth, and hence the largest possible number of terminals is (9600/15)=640 terminals. In fact, we can only utilize the channel 85.3% at most, and so the largest number of terminals this could support is 85.3% of 640 terminals, or 545 terminals. How do we interpret this number? If we never have more than two hundred terminals actively using the system, this is fine; most terminals will simply transmit a keystroke and receive an acknowledgement, with little load on the link. If we have one thousand or more terminals simultaneously using the system, most terminals will experience unacceptable delays.

11.8.2 Slotted Operation with Loss In a slotted system there is a common clock or measure of time for all stations. Time is broken up into equal length time intervals called time slots, and all attempts are made at the start of a slot. For this case,

$$\max U = \frac{E(T_M)}{E(T_M) + T_O(e-1)}$$

The interpretation of this is that under maximum link utilization, there are

$e-1 \approx 1.71$ collisions for every successful message transmission For the space satellite application with $E(T_M) = T_O$, we see

$$\max U = \frac{1}{e} \quad E(T_M) = T_O$$

In our example of terminals generating fifteen bits per second of traffic over a 9600 bps link, with a 50 μ second overhead time, we see that the maximum link utilization can be increased to

$$\max U = \frac{15\ bits/9600\ bps}{15\ bits/9600\ bps + 50\ \mu sec\,(e-1)} = 94.2\%$$

The maximum number of terminals the link can support is increased from 545 with unslotted or unsynchronized operation to 603 with slotted or synchronized operation. Is it worth it? This must be addressed relative to many other factors: link access performance is only one consideration.

11.8.3 Aloha Mean Throughput Rate Analysis Here is an analysis based on a different set of *assumptions* that leads to the same conclusions as the previous section.

Messages arrive to be transmitted over a serially reusable link according to simple Poisson arrival statistics, with mean arrival rate λ. The sequence of message transmission times are independent identically distributed random variables drawn from a common distribution $G_{T_M}(X)$ with $\hat{G}_{T_M}(z)$ denoting the associated moment generating function $E[exp(-zT_M)]$. If a message transmission attempt is made, and no other attempts are made for the duration of the message transmission, then the message transmission is successful. If two or more attempts to seize the channel are made during a message transmission, all attempts are unsuccessful or lost and will presumably retry later.

The fraction of time the transmission medium is idle is denoted by π_0, the fraction of time no message transmission attempts are being made, which must equal

$$\pi_0 = e^{-\lambda E(T_M)}$$

On the other hand, the mean transition rate out of this idle state, denoted R_0, multiplied by the mean sojourn time in this state, denoted $1/\lambda$, must equal the fraction of time the transmission medium is idle:

$$\pi_0 = \frac{R_0}{\lambda}$$

Next, we observe that the transition rate out of the idle state must equal the transition rate into the state where one message is being transmitted, denoted R_1:

$$R_1 = R_0$$

Finally, the rate of successfully transmitting messages is simply the rate of entering state of one active transmission and no other messages arriving during that transmission, denoted $\hat{G}_{T_M}(\lambda)$

$$R_S = R_1 \hat{G}_{T_M}(\lambda)$$

Combining all this, we see

$$R_0 = R_1 = \lambda e^{-\lambda E(T_M)}$$

$$R_S = \lambda e^{-\lambda E(T_M)} \hat{G}_{T_M}(\lambda)$$

For the special case where the message transmission time distribution is deterministic, we see

$$R_S = \lambda e^{-2\lambda E(T_M)}$$

If we maximize the mean throughput rate over the message arrival rate, we find that

$$\max_{\lambda} R_S \approx \frac{1}{2e} \frac{1}{E(T_M)}$$

which is what we found in the earlier section, under different assumptions.

11.8.4 Additional Reading

[1] N.Abramson, *The Throughput of Packet Broadcasting Channels*, IEEE Transactions on Communications, **25**, 117-128 (1977).

[2] G.T.Almes, E.D.Lazowska, *The Behavior of Ethernet-Like Computer Communications Networks*, Proceedings of Seventh Symposium on Operating Systems Principles, pp.66-81, Asilomar Grounds, Pacific Grove, California, 10-12 December 1979.

[3] D.R.Boggs, R.M.Metcalfe, *Ethernet: Distributed Packet Switching for Local Computer Networks*, Communications ACM, **19** (7), 395-404 (1976).

[4] A.B.Carleial, M.E.Hellman, *Bistable Behavior of ALOHA-Type Systems*, IEEE Transactions on Communications, **23** (4), 401-410 (1975).

[5] M.Kaplan, *A Sufficient Condition for Nonergodicity of a Markov Chain*, IEEE Transactions on Information Theory, **25** (4), 470-471 (1979).

[6] L.Kleinrock, S.S.Lam, *Packet Switching in a Multiaccess Broadcast Channel: Performance Evaluation*, IEEE Transactions on Communications, **23** (3), 410-422 (1975).

[7] O.Spaniol, *Analysis and Performance Evaluation of HYPERchannel Access Protocols,* Performance Evaluation, **1**, 170-179 (1981).

[8] F.Tobagi, *Multiaccess Protocols in Packet Communication Systems,* IEEE Transactions on Communications, **28** (4), 468-488 (1980).

11.9 Performance Analysis of Slotted P-Persistent Carrier Sense Access

Here is an analytically tractable model for CSCD that bounds not only mean throughput rate but also delay statistics.

11.9.1 Model A set of stations are connected via a bus. Each station continuously senses the transmission medium to see if carrier is present. A clock synchronizes the actions of all stations, with the clock period called a time slot. A time slot is chosen equal to the worst case time for energy to propagate from one station to another plus allow all electronic transients to decay to acceptable levels. All transmission attempts begin at the start of a time slot, and all messages transmitted are an integral number of time slots. If a station has a message to transmit, it first attempts to seize the transmission medium by transmitting a preamble equal to one time slot in duration; if the preamble is transmitted without distortion (e.g., due to another station transmitting its preamble, which is called a collision), then the station proceeds to transmit its message. If the preamble is distorted, then a collision is said to occur, and each station involved in the collision will retry at a later point in time to go through the same process. In a local network, the preamble transmission time or time slot is typically small compared to the message transmission time. As the transmission speed increases, the slot time or time required to resolve contention becomes relatively more important compared to the mean message transmission time. The retry policy adopted here is called *p-persistent* because if station becomes active, and the transmission medium is busy (with either a message transmission or a collision), the station will wait (or persist) until the transmission medium becomes idle. The parameter p denotes the probability that a station involved in a collision will retry in the next time slot, or with probability $(1-p)$ repeat this retry process in the subsequent time slot.

Messages arrive according to a very particular pattern which is practical to implement as a test procedure or diagnostic in such systems: all but one of the stations always have a message to transmit. The remaining station generates message transmission attempts according to Poisson statistics. This will bound the delay for any one station, i.e., this is a *worst* case analysis. Furthermore, this analysis is *sharp* or *achievable,* and hence provides a simple check on operations.

11.9.2 Message Arrival Statistics N stations are connected to a common transmission medium or bus. A clock signal with period T_{slot} is transmitted over the bus. Every station but one, i.e., a total of $(N-1)$ stations, always has a message to transmit. The final station has messages arrive for transmission according to simple Poisson arrival statistics with mean arrival rate λ.

11.9.3 Message Length Statistics The message length statistics are identical for all stations, with the message lengths forming a sequence of independent identically distributed random variables. The message transmission time T_M has moment generating function $E[e^{-zT_M}]$ given by

$$E[e^{-zT_M}] = \sum_{K=1}^{\infty} F(K) e^{-zKT_M} \quad \sum_{K=1}^{\infty} F(K) = 1, 0 \leq F(K) \leq 1$$

$F(K), K=1,...$ is the fraction of messages requiring $K \geq 1$ time slots to be completely transmitted.

11.9.4 Station Buffering Policy Each station can buffer an infinite number of messages. Messages are transmitted in order of arrival from each station.

11.9.5 Transmission Medium Access Policy The transmission medium is a serially reusable resource. Contention for this shared resource is arbitrated as follows:

- All stations sense the state of the transmission medium at all times to see if it is busy
- If the transmission medium is not busy, then with probability p a station that has a message to transmit will attempt to transmit in the next time slot, and with probability $1-p$ will wait until the subsequent time slot and repeat this process
- If one station successfully seizes the transmission medium for one time slot, it will hold it for the duration of the message transmission
- If two or more stations attempt to seize the transmission medium during one time slot, neither will succeed, and both will retry according to the above policy

11.9.6 Goals We wish to calculate the mean throughput rate and delay statistics of D, the message delay statistics encountered by messages at one station, with all the other stations always having a message to transmit.

11.9.7 Analysis The probability that a given station is successful in seizing the transmission medium, given that a total of K stations actually have messages to transmit is given by Q_K:

$$Q_K = PROB[a \text{ given station successfully seizes a time slot}] = p(1-p)^{K-1}$$

The sequence of time intervals that the server is absent (handling message transmissions and collisions) are independent identically distributed arbitrary random variables called intervisit times. The random variable for the delay in this model is the sum of three random variables

[1] The time interval from when a message arrives at a station until the local area network becomes idle, denoted $\tilde{V}$, the forward recurrence time of the intervisit time

[2] W, the time spent waiting to transmit all messages queued in the buffer of a station that arrived ahead of a given arrival

[3] $\tilde{T}_M$, the time measured from the start of the first transmission attempt until the message is successfully transmitted; this includes the time waiting for collisions and for other successful attempts by other stations attempting to use the channel, plus the time to actually transmit the message, T_M

Symbolically, this is written as

$$D = \tilde{V} + W + \tilde{T}_M$$

All of these random variables on the right hand side are statistically independent of one another:

- The forward recurrence time of the intervisit distribution depends on the load at the other stations and the arrival statistics of messages at the final station; the other stations always have a message to transmit
- The waiting time to transmit messages ahead of a given arrival depends only on the number of messages ahead of a given arrival, and not on the intervisit time statistics or the message transmission statistics of the given arrival
- The transmission time statistics depend only on the activity of the other stations and the access method, not on the intervisit statistics or waiting time statistics

The moment generating function of D is the product of the individual moment generating functions:

$$E[e^{-zD}] = E[e^{-z\tilde{V}}]E[e^{-zW}]E[e^{-z\tilde{T}_M}]$$

We now calculate each of these three individual moment generating functions.

11.9.8 Forward Recurrence Time Distribution Moment Generating Function The moment generating function for the intervisit time V follows from decomposing an intervisit into two distinct events with different probabilities:

[1] The transmission medium is busy transmitting a message from one of the other $(N-1)$ stations

[2] The transmission medium is busy with a collision between two or more stations

The moment generating function of the intervisit time distribution is given by

$$E[e^{-zV}] = (N-1)Q_{N-1}E[e^{-zT_M}] + [1-(N-1)Q_{N-1}]e^{-zT_{slot}}$$

The moment generating function for the forward recurrence time distribution is given by

$$E[e^{-z\tilde{V}}] = \frac{1-E[e^{-zV}]}{zE(V)}$$

11.9.9 Waiting Time Distribution Moment Generating Function The waiting time moment generating function is given by

$$E[e^{-zW}] = \frac{z[1-\lambda E(\tilde{T}_M)]}{z-\lambda[1-E(e^{-z\tilde{T}_M})]} \quad \lambda E[\tilde{T}_M] < 1$$

In words, we demand that the mean arrival rate of messages be less than the total time required to successfully transmit a message $(\lambda E(\tilde{T}_M) < 1$.

11.9.10 Moment Generating Function of Inflated Transmission Time Distribution The final moment generating function, for the message transmission time, involves taking into account three distinct events:

[1] With probability Q_N the last station will succeed and hold the transmission medium for an interval of duration T_M

[2] With probability $(N-1)Q_N$ one of the other $(N-1)$ stations will succeed and hold the transmission medium for an interval of duration $\tilde{T}_M$ and then the last station will succeed and hold the transmission medium for an interval of duration T_M

[3] With probability $1-NQ_N$ there will be a collision consuming a time slot of duration T_{slot} followed by a time interval of duration $\tilde{T}_M$.

Combining all these terms, the moment generating function for $\tilde{T}_M$ is given by

$$E[e^{-z\tilde{T}_M}] =$$

$$Q_N E[e^{-zT_M}] + (N-1)Q_N E[e^{-zT_M}]E[e^{-z\tilde{T}_M}] + [1-NQ_N]E[e^{-z\tilde{T}_M}]e^{-zT_{slot}}$$

$$E[e^{-z\tilde{T}_M}] = \frac{Q_N E[e^{-zT_M}]}{1-(N-1)Q_N E[e^{-zT_M}] - [1-NQ_N]e^{-zT_{slot}}}$$

The mean or first moment of the delay process is given by

$$E(D) = E(\tilde{V}) + E(W) + E(\tilde{T}_M)$$

$$E(\tilde{V}) = \frac{(N-1)Q_{N-1}E(T_M^2) + T_{slot}^2[1-(N-1)Q_{N-1}]}{2(N-1)Q_{N-1}E(T_M) + 2T_{slot}[1-(N-1)Q_{N-1}]}$$

$$E(W) \quad \frac{\lambda E(\tilde{T}^2_M)}{2(1-\lambda E(\tilde{T}_M))}$$

$$E(\tilde{T}_M) \quad NE(T_M) \cdot \frac{T_{slot}(1-NQ_N)}{Q_N}$$

$$E(\tilde{T}^2_M)$$

$$NE(T^2_M) + (N \quad)E(T_M)E(\tilde{T}_M) + \quad \frac{NQ_N}{Q_N}[T^2_{slot}+2T_{slot}E(\tilde{T}_M)]$$

There are two regions of operation:

- Light loading, $\lambda \rightarrow 0$, where the mean delay is dominated by the mean of the forward recurrence time of the intervisit time plus the mean of the inflated message transmission time

$$E(D) \approx E(\tilde{V}) + E(\tilde{T}_M) \quad \lambda \rightarrow 0$$

- Heavy loading, $\lambda \rightarrow 1/E(\tilde{T}_M)$, where the mean delay is dominated by the mean waiting time to transmit messages

$$E(D) \approx E(W) \quad \lambda \quad E(\tilde{T}_M)$$

$E(\tilde{T}_M)$ equals a term proportional to each station transmitting a message plus a term that is dependent on controlling access to the channel. The mean or first moment of message delay depends not only on the *first* but also the *second* moment of the message transmission time distribution. Thus, knowing only the mean message length only allows the maximum mean throughput rate to be determined, but *nothing* can be said concerning mean delay unless information about fluctuations about the mean message transmission time is available.

There is still one degree of freedom left: how do we choose p? One way to choose p is to maximize the mean throughput rate, i.e., to choose p such that we have the largest mean arrival rate to one queue that still results in a nondegenerate statistical equilibrium distribution. This results in choosing $p = 1/N$. In words, each station that is competing for access is equally likely to gain control of the transmission medium when it becomes available. For sufficiently many stations, $N >> 1$, the condition for statistical equilibrium becomes

$$N\lambda < \frac{}{E(T_M)+T_{slot}(e-1)} \quad N >> .$$

A different way of interpreting this result is to consider the channel in one of two states. The first state is due to arbitration or controlling access, and has mean duration $(e-1)T_{slot}$. The second state is due to successful data transmission, and has mean duration $E(T_M)$. The maximum mean throughput rate, which we identify with $N\lambda$, is the reciprocal of the sum of the mean time spent in each state. The mean access time is controlled by the worst case propagation of energy from one end of the transmission medium to the other, while the mean message transmission time is controlled by the mean message size (measured in bits) and the data transmission rate (measured in bits per second). One way to understand the interaction of these parameters is to see when $(e-1)T_{slot}$ equals $E(T_M)$, because at this point half of the time is spent in message transmission and half is spent in controlling channel access. A different type of insight is that for high maximum mean throughput rate we wish to design the system so that little time is spent in controlling access and most of the time is spent in data transmission, i.e., $E(T_M)$ must be much greater than $(e-1)T_{slot}$. This can occur by increasing the message length (in bits) or decreasing the data transmission rate (in bits per second), holding all other parameters fixed.

11.9.11 Illustrative Numerical Studies We now turn to an illustrative case study to numerically explore design consequences.

11.9.12 Parameters The model ingredients are

- Arrival statistics
  - Number of stations--20, 40, 60, 80, 100 stations
  - Arrival rate of messages at each station--All but one station always has a message, while the final station has messages arriving according to Poisson statistics with rate λ
- Message length statistics
  - Message length distribution (in bits)-- Exponentially distributed and constant length messages, with means of 500, 1000, and 2000 bits per message
  - One way propagation time of energy from one end to the other-- ten or twenty microseconds
  - Transmission rate, measured in bits per second-- One, three or ten megabits per second

11.9.13 Maximum Mean Throughput Rate The figures below plot the carried data transmission rate versus the actual data transmission rate for these parameters. If the channel were used for data transmission alone, this would be a straight line with slope one; since the channel is used for control and data transmission, the result is a curve that lies below the zero overhead straight line. The distance from the straight line zero overhead case to the actual curve shows the amount of transmission capacity devoted to controlling channel access.

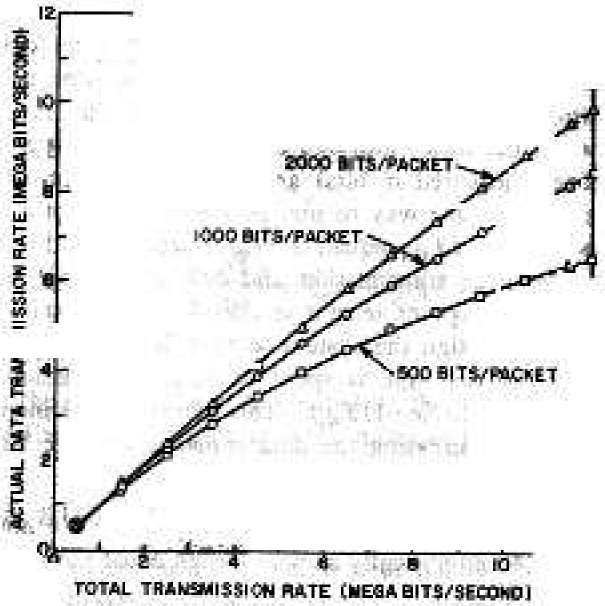


Figure 11.17.Maximum Mean Throughput Rate vs Clock Rate

11.9.14 Message Delay Statistics Next, we fix the mean fraction of time the transmission medium is busy transmitting data, which is called the *utilization*, and equals the mean arrival rate of messages multiplied by the mean message transmission time; since the mean message length is fixed, this fixes the mean arrival rate. The figures below plot illustrative upper bounds on mean delay versus line utilization due to data transmission alone for the numbers described above. The parameter p is chosen to the reciprocal of the number of stations in all cases.

For example, with fifty per cent utilization, one hundred stations, ten megabits transmission speed, one thousand bits per constant length message, ten microseconds one way propagation time (one hundred bit transmission times), the mean delay is upper bounded by over 1.4 seconds. If we halve the message length to five hundred bits, the upper bound on mean message delay does *not* halve, but drops to 0.9 seconds; if we double the message length to two thousand bits, the upper bound on mean message delay does *not* double, but increases to 2.3 seconds. If we increase the slot time from ten to twenty microseconds, with one thousand bits per constant length message, and a transmission rate of ten megabits per second, the upper bound on mean message delay increases to 1.8 seconds. Finally, changing the shape of the message length distribution by going from constant to exponentially distributed message lengths, has negligible impact on the upper bound on mean delay.

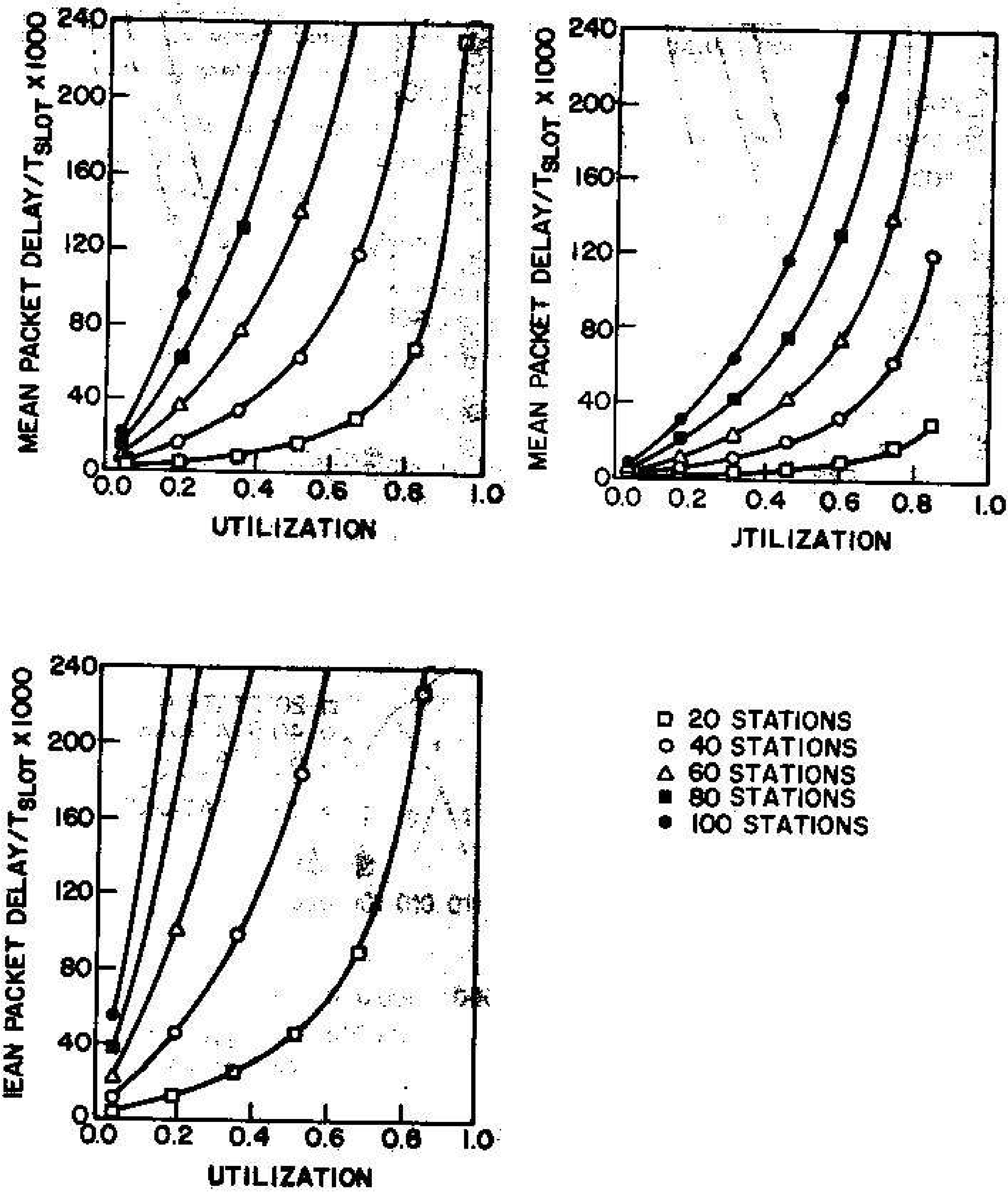


Figure 11.18.Mean Message Delay Upper Bound vs Utilization
1000 Bits/Packet; Constant Packet Size; $T_{prop}=10\mu sec$
Lower Left: 1 MBPS; Upper Left: 3 MBPS; Upper Right: 10 MBPS

11.9.15 Interpretation **These** numerical plots quantify the following phenomena:

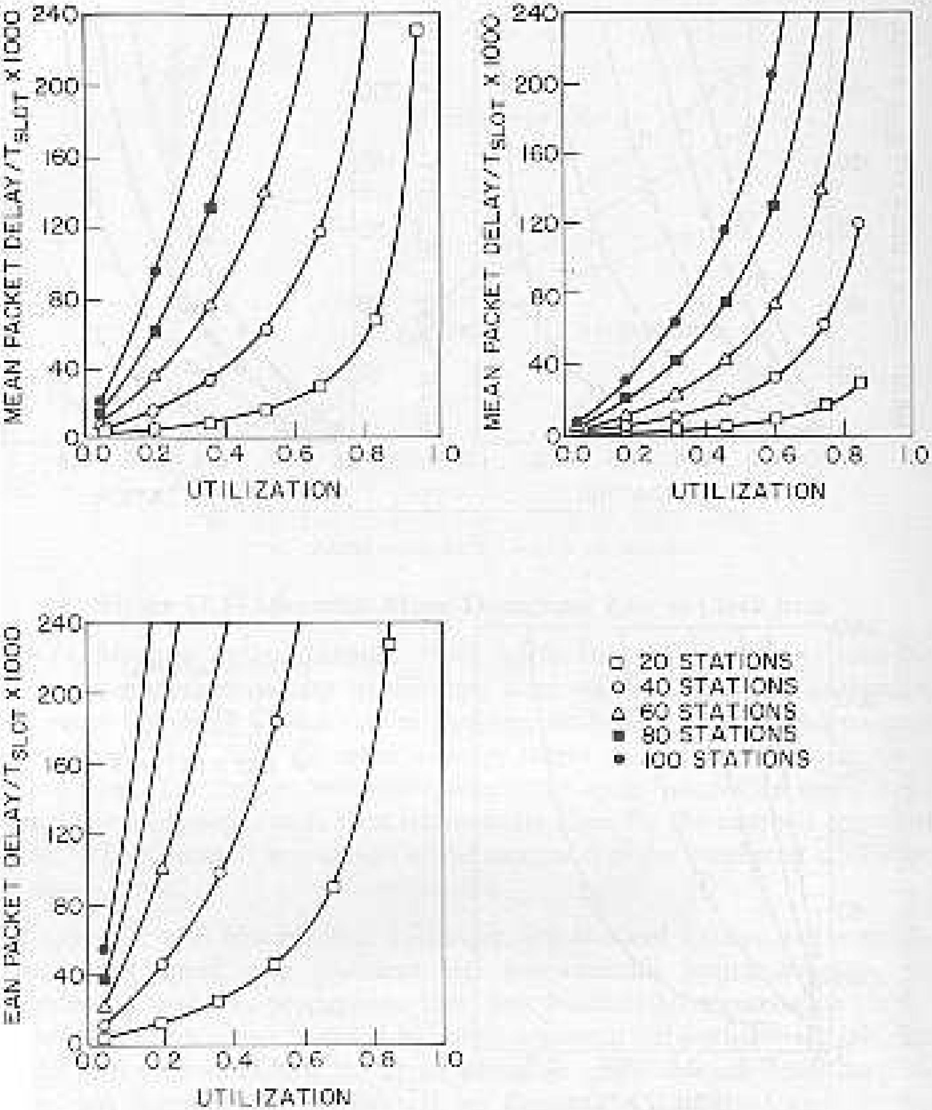


Figure 11.19. Mean Message Delay Upper Bound vs Utilization
1000 Bits/Packet; Exponential Packet Size Distribution; $T_{prop}=10\mu sec$
Lower Left: 1 MBPS; Upper Left: 3 MBPS; Upper Right: 10 MBPS

As the slot time increases, with all other parameters fixed, the mean delay upper bound increases and the maximum bandwidth available for data transmission decreases.

- As the mean packet length increases, with all other parameters fixed, the mean delay upper bound increases and the maximum bandwidth available for data transmission increases.
- As the fluctuations about mean packet length increase from constant to exponential distribution, the mean delay upper bound increases and the maximum bandwidth available for data transmission is not affected.
- As more and more stations are added, with all other parameters fixed, in particular with the mean utilization of the link due to data transmission fixed, the maximum bandwidth available for data transmission is not affected, but the mean delay upper bound increases.

11.10 Reservation Decision Tree Access via a Bus

Reservation decision tree access is the least studied access method of the proposed bus access methods. The figure below shows illustrative operation of a reservation decision tree access policy, where the station addresses are used to determine retry priorities for stations involved in message collisions.

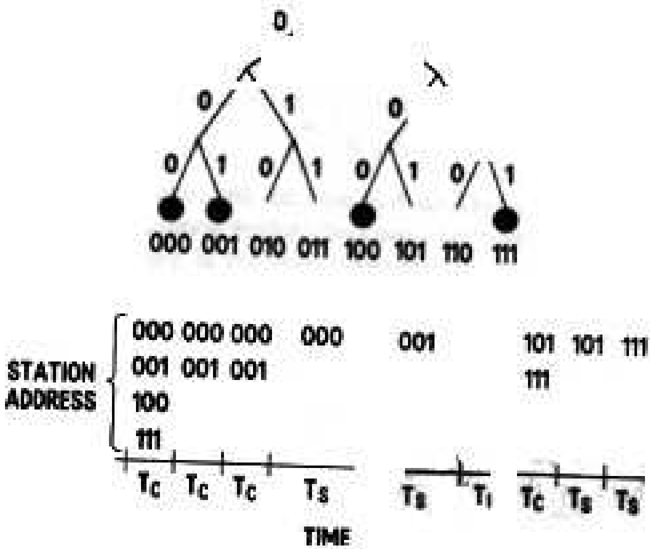


Figure 11.20.Reservation Decision Tree Access

An illustrative scenario begins at the start of a frame, when all stations that have a message to transmit go active and collide. The retry policy involves a controlled priority arbitration among all stations active at the start of the frame, with the station addresses used to determine or control the retry priorities.* After the first collision, only those stations whose most significant address bit is a zero can retry, and all other stations are silent: stations zero and one will attempt to transmit in the next time slot, and stations four and seven will defer transmission attempts. Once more, stations zero and one transmit during the same time slot, and now all stations whose first two leading address bits are zero will attempt to transmit in the next slot, and all other stations will defer: stations zero and one will attempt to transmit, while stations four and seven will defer transmission attempts. Once more stations zero and one transmission attempts collide in the next time slot, and now only those stations with three leading address bits all zero will attempt to transmit in the next time slot: only station zero will retry, and stations one, four and seven will defer. In the next time slot, station zero succeeds in transmitting. Now we allow all stations whose leading address bits are two zeroes followed by a one to transmit in the next time slot: station one will transmit then, with stations four and seven deferring. Now all stations with a leading zero and one address bit are allowed to transmit in the next time slot: this would be stations two and three, neither of which has a message, and hence the time slot is an idle time slot. All stations whose address begins with a zero have been allowed to transmit and have successfully transmitted a message, if they had one at the start of the frame. Now we repeat the process: all stations with a leading address bit of one, stations four and seven, attempt to transmit in the next time slot, and their attempts collide. In retrying, all stations with a leading address bit sequence of one followed by zero, which is only station four, attempt to transmit in the next time slot: station four is successful. Finally, all stations with leading address bits of one followed by one, which is only station seven, attempt to transmit in the next time slot: station seven is successful. The figure is a graphical summary of this process.

Why does this look like carrier sense collision detection under light load? Because if only one station has a message, it will transmit it immediately. Why does this look like token passing or polling under heavy load? Because if every station has a message, we will have 2^N-1 collisions for 2^N stations, hence we will effectively pass a token 2^N-1 times among 2^N stations. The key observation here is that we wish to interrogate groups of stations once a collision occurs, with at most one station in each group having a message. By controlling the retry process, making it quite predictable and regular, we gain in traffic handling capability over irregular retry and retransmission.

* Those familiar with algorithms will recognize this as a topological binary tree sort, with the nodes of the tree being searched depth first, and then left to right; station retry priorities determine the graph.

The figure below shows illustrative operation of a related policy, where the messages are transmitted in order of arrival using the message arrival time to determine priority ordering; the next figure shows the associated decision tree used to arbitrate contention.

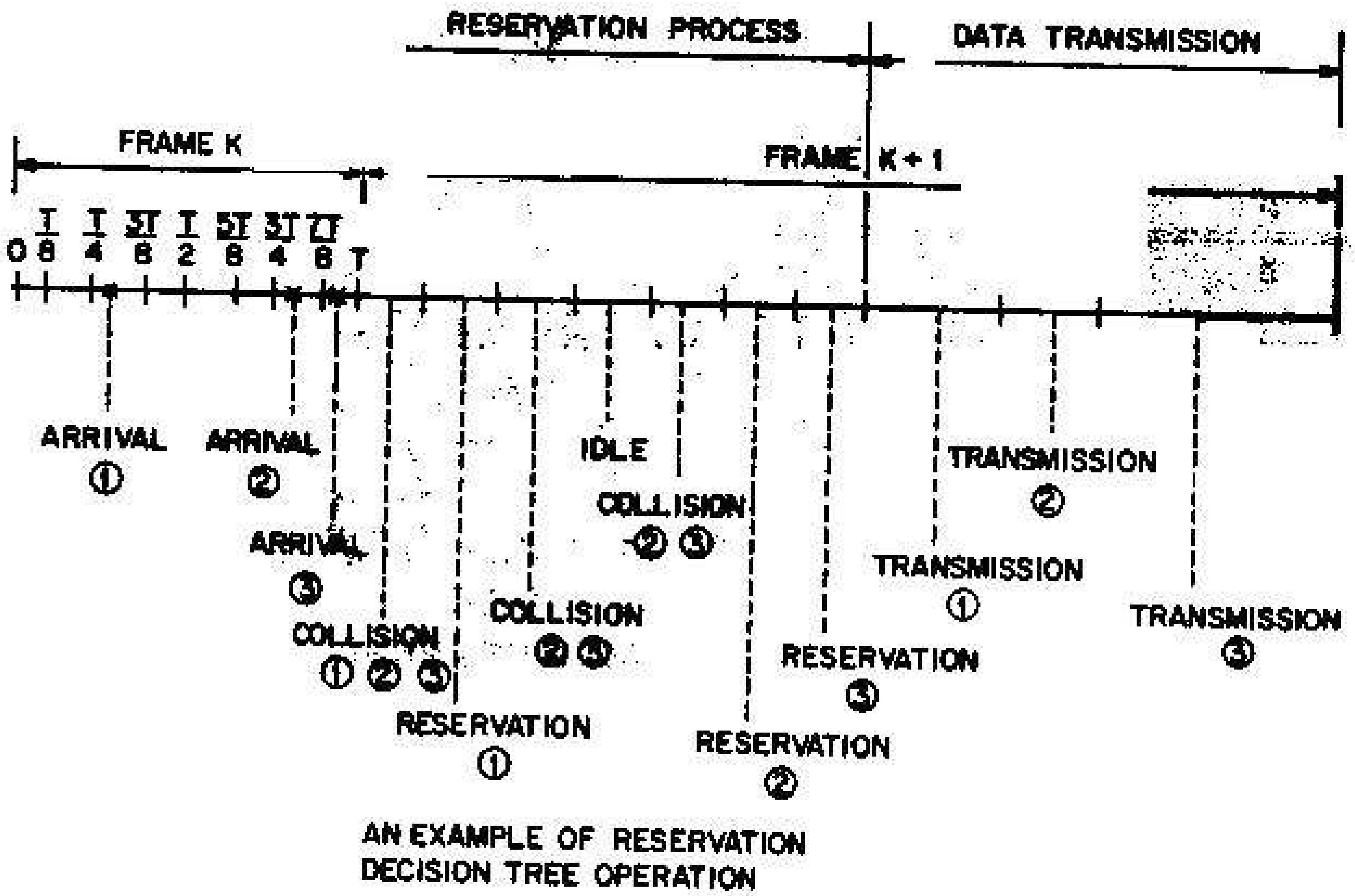


Figure 11.21.Reservation Tree Access Order of Arrival Transmission

For this example we assume all stations have a clock available for determining the duration of each frame, unlike the previous example. The parameter T shown in the figure above is the duration of the *previous* frame: each station uses its clock to determine if it became active during the previous frame or not, and then each station refines this to determine in which half of the previous frame it became active, and then iterates. These are just two examples of the many possible policies that can be realized by this policy class.

Here are highlights of reservation access:

- Transceivers are logically organized into a tree, with the leaves of the tree determining the urgency of message transmission.
- After a collision, all messages involved in a collision are successfully transmitted before any subsequent arrivals are transmitted.
- All of the traffic handling hooks in polling can still be used (multiple visits per logical frame, maximum number of frames transmitted per visit, dedicating transmission capacity to each service).

We stress the following points:

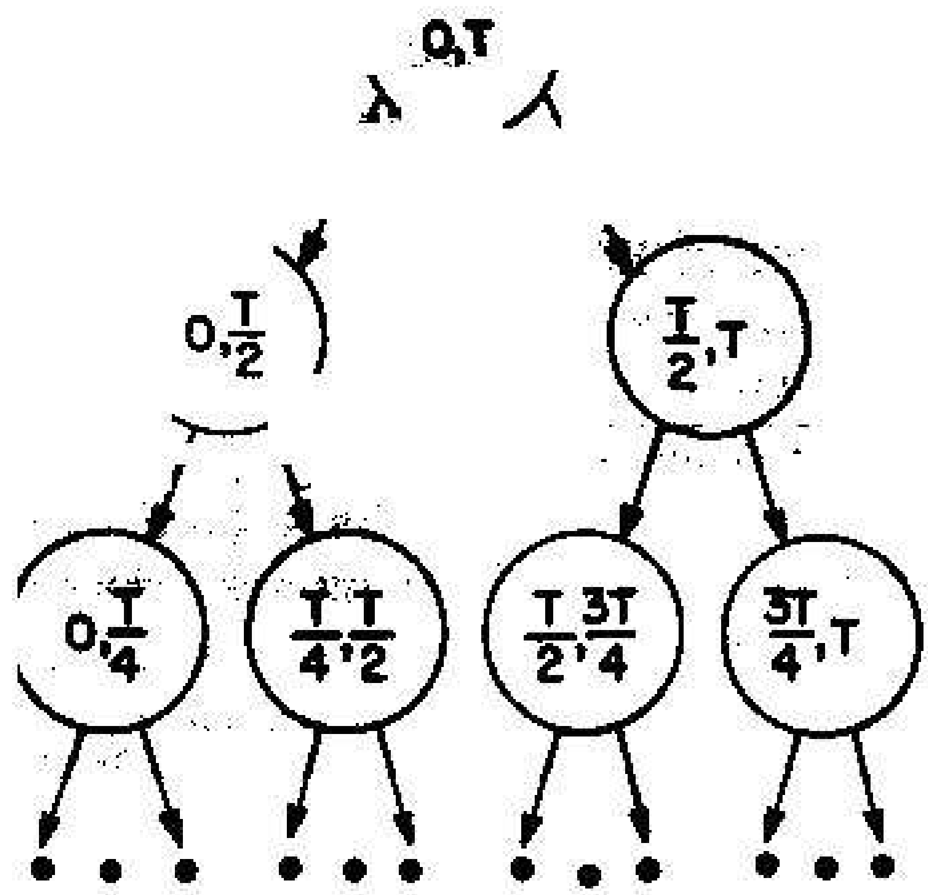


Figure 11.22.Decision Tree for Order of Arrival Transmission

- Mean time required to resolve collisions per transmission is bounded by a logarithmic function of the number of ready transceivers (here for eight ready stations in the finite tree we need on the order of three collisions to the first successful transmission).

- Maximum time required to resolve collisions per transmission is bounded; the example for the finite tree shows that for eight stations at most seven collisions occur.

  Mean waiting is linearly proportional to the number of active stations: each station active at the start of each frame is guaranteed of being able to successfully transmit its message within each frame, and there are at most 2^N stations.

- There is roughly one control overhead time slot interval for every one successful message transmission, just as in central control (cf. the above examples). What is more subtle is that the variance of the overhead time per successful message transmission only grows *linearly* with the number of stations attempting to transmit at the start of a frame, just as in central control.

If we return to the first example, we see that an additional degree of freedom is available: at what priority level do we attempt to start our retry arbitration: using the most significant address bit of each station, or the two most significant address bits of each station, or so forth? One way to quantify our intuition on this is to examine the number of steps required (idle, collision, and

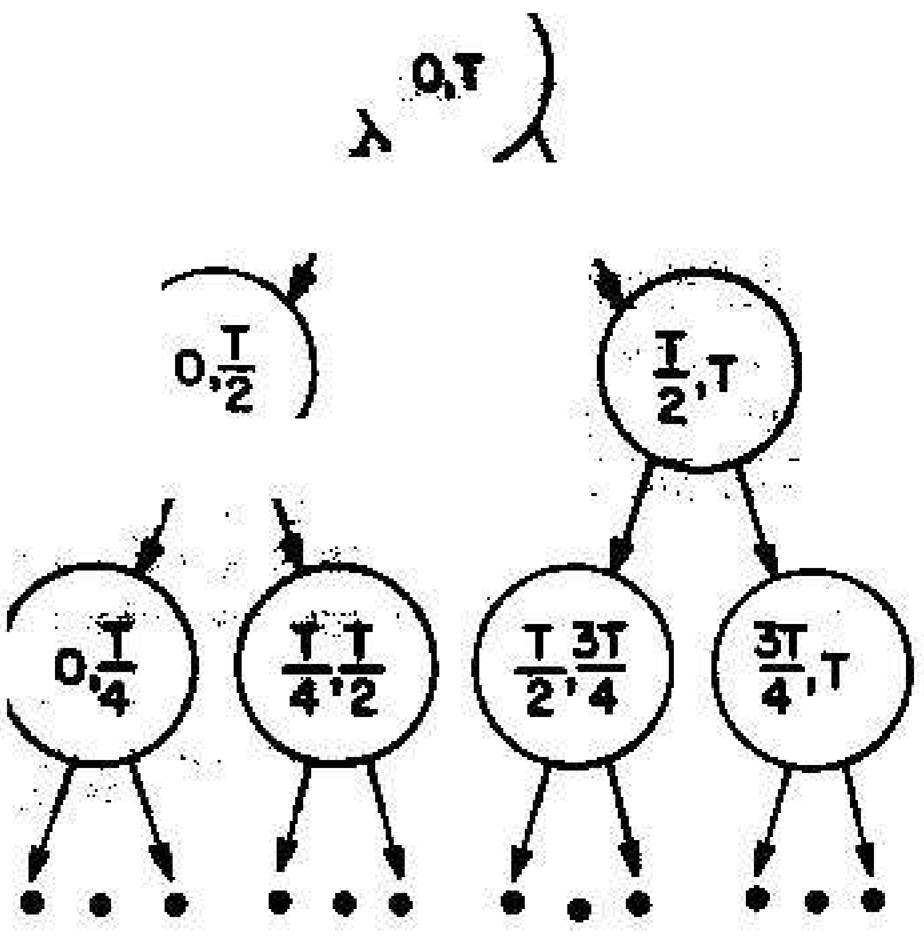


Figure 11.22.Decision Tree for Order of Arrival Transmission

- Mean time required to resolve collisions per transmission is bounded by a logarithmic function of the number of ready transceivers (here for eight ready stations in the finite tree we need on the order of three collisions to the first successful transmission).

- Maximum time required to resolve collisions per transmission is bounded; the example for the finite tree shows that for eight stations at most seven collisions occur.

  Mean waiting is linearly proportional to the number of active stations: each station active at the start of each frame is guaranteed of being able to successfully transmit its message within each frame, and there are at most 2^N stations.

- There is roughly one control overhead time slot interval for every one successful message transmission, just as in central control (cf. the above examples). What is more subtle is that the variance of the overhead time per successful message transmission only grows *linearly* with the number of stations attempting to transmit at the start of a frame, just as in central control.

If we return to the first example, we see that an additional degree of freedom is available: at what priority level do we attempt to start our retry arbitration: using the most significant address bit of each station, or the two most significant address bits of each station, or so forth? One way to quantify our intuition on this is to examine the number of steps required (idle, collision, and active (A) or idle (I) for the next step

Table 11.9. Station States at Conclusion of Step in Decision Tree

Step	Station [illegible]			Station [illegible]			Station [illegible]			Station [illegible]		
Number	Frame	Count	All	Frame	Count		Frame	Count	All	Frame	Count	All
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

At each step, each station consults its priority index (the station address for the first example, the arrival time within the frame for the second example) and its transmission control and frame synchronization counter state. If a station becomes active during a frame, it will defer transmission attempts until the start of the next frame, when its frame synchronization counter strikes zero.

11.10.1 Decision Tree Search Access Performance Analysis Here are the results of a traffic analysis for the decision tree arbitration policy.

First we must specify the arrival statistics:

- The message interarrival times are independent identically distributed exponential random variables
- Message lengths are independent identically distributed arbitrary random variables with associated moment generating function $\hat{G}_M(x)$

Next the access method must be specified:

- Basic transmission time unit is a time slot. A frame is made up of a variable number of time slots
- Messages that arrive during a frame will be transmitted during the next frame
- Initial time slots within a frame are devoted to determining which sources have a message to transmit
- Final time slots within a frame are devoted to data transmission from active sources

The process $L_N, N>0$ denotes the number of time slots consumed for reservation by N sources at the start of a frame. Its moment generating

station address for the
econd example) and its
er state. If a station
ion attempts until the
ounter strikes zero.

nalysis Here are the

identically distributed

uted arbitrary random
$\hat{G}_M(x)$

rame is made up of a

mitted during the next

rmining which sources

ransmission from active

ne slots consumed for
Its moment generating

function is given by

$$\eta_N(x \quad E[x^{L_N}]$$

It can be shown that

$$\lim_{N\to\infty}\frac{E(N)}{N} \qquad \frac{2}{1 \quad [illegible]} \approx 2.88542.$$

$$\text{if } \lambda < \overline{\alpha+E(T_{message})}$$

The mean message delay, including transmission plus waiting, is given by

$$E(T_{delay}) - E(T_{message}) + \frac{1+\lambda E(T_{message})}{2\lambda} \times$$

$$\frac{\sum_{K=1}^{\infty} K(K \quad 1)\pi_K}{\sum_{K=1}^{\infty} K\pi_K} + \frac{\sum_{K=1}^{\infty}\pi_K K E(L_K)}{\sum_{K=1}^{\infty} K\pi_K}$$

Note that the mean delay involves first and second moments of L_N. In some cases it may prove easier to work with upper and lower bounds on the first and second moments rather than the statistics of L_N. The tightest possible bounds known to the authors are

$$\underline{a}\ K + \underline{b} \leqslant E(L_K) \leqslant \bar{a}\ K + \bar{b}$$

$$\underline{c}\ K^2 + \underline{d}\ K + \underline{e} \leqslant E(L_K{}^2) \leqslant \bar{c}\ K^2 + \bar{d}\ K + \bar{e}$$

The best known coefficients for these bounds at the present time are

$$2.8810K - 1 \leqslant E(L_K) \leqslant 2.28965K - 1$$

$$8.300K^2 + 0.4880K + 1 \leqslant E[L_K(L_K - 1)]$$

$$\leqslant 8.333K^2 + 0.5175K + 1$$

These expressions have been found to be useful in practice for rough sizing of system performance and to guide refinements of this analysis.

11.10.2 Additional Reading

[1] E.H.Frei, J.Goldberg, *A Method for Resolving Multiple Responses in a Parallel Search File*, IRE Transactions on Electronic Computers, **10** (4), 718-722 (1961).

[2] N.Pippenger, *Bounds on the Performance of Protocols for a Multiple-Access Broadcast Channel*, IEEE Transactions on Information Theory, **27** (2), 145-151 (1981).

[3] I.Rubin, *Synchronous and Channel-Sense Asynchronous Dynamic Group-Random-Access Schemes for Multiple-Access Communications* IEEE Transactions on Communications, 31 (9) 1063-1077 (1983).

[4] R.Seeber, A.B.Lindquist, *Associative Memory with Ordered Retrieval*, IBM J.Research and Development, 6 (1), 126-136 (1962).

11.11 Token Passing Access via a Ring

Token passing access via a ring is logically identical to that for a bus. We will only dwell on differences here between a bus and a ring. The physical topology of a ring is that each station transmits to only one station and receives from only one station. The token is a unique bit pattern that circulates around the ring. When it is transmitted to a station ready to send a packet, that station changes the token bit pattern on the fly, i.e., destroys the token, and then appends its packet. The packet circulates around the ring, to the intended receiving station, and then returns to the transmitting station to acknowledge complete successful transmission around the ring. Because of the geographic extent of the local area network, at most *one* packet can be circulating about the ring at any one time.

Since this is a variant on polling, a number of traffic load balancing mechanisms are available. *Separation or dedication* of transmission capacity to different types of services is still possible. The number of visits per polling cycle per station can be controlled, and the maximum number of frames transmitted per visit can be set. Overhead time per transmission is bounded by a linear function of the number of transceivers (typically this is a constant due to ring propagation and circuitry transients plus a linear term due to the transceiver processing) because control must be passed to each station at least once in a ring or polling cycle; this will typically be *much* smaller for a ring than for a bus, because the propagation is now only from station to station, not worst case from one end of the bus to the other, and because the transceiver token processing can be done in parallel (except for a delay on the order of one bit transmission time per interface). Typical numbers for token passing from station to station over a ring are one microsecond, while a bus requires ten times this typically. This improves delay access under light load, where typically we must wait for the token to be passed through half the stations, and it improves the maximum mean throughput rate under heavy load, since less time is spent in token passing versus message transmission. High throughput under heavy load is achieved because most of the time the transmission medium will be busy transmitting messages under load, not passing the token from station to station. Mean message delay time that is linear in the number of stations is achievable because if each station has a message to transmit then the waiting time at any one station is simply the time required for all other stations in the polling cycle to transmit.

11.11.1 Additional Reading

[1] W.D.Farmer, E.E.Newhall, *An Experimental Distributed Switching System to Handle Bursty Computer Traffic*, Proc.ACM Symp.Problems in the Optimization of Data Communication Systems, pp.1-34, Pine Mountain, GA, October 1969.

11.12 Comparisons

What types of questions do we wish to answer in comparing different access methods? First, let's deal with mean value measurements and statistics, before turning to more sophisticated second order statistics involving fluctuations and correlation:

- What fraction of transmission capacity is devoted to control versus data transmission?
- What is the impact on transmission speed versus throughput and delay?
- How sensitive is system performance to a change in workload, such as changing the message length, or having voice and data services instead of just data?
- How do protocol parameters (length of header, carrier sense collision detection jam time or interframe gap time) influence traffic handling characteristics?

Two workloads are used to bound performance. In Figure 11.23, one station out of one hundred is always actively transmitting a message, while in Figure 11.24 one hundred out of one hundred stations are active attempting to transmit messages. All messages are four thousand bits long. The horizontal axis or abscissa in either case is the raw data transmission speed of the network (the clock rate), while the vertical axis or ordinate is the actual carried data rate, the rate at which data bits are successfully transmitted. The ideal case is to use zero transmission capacity for network access control, which would be a straight line with slope unity. The deviation from this straight line shows the penalty paid using the same network to control access as well as transmit messages.

Figure 11.25 is a plot of lower bound on message delay versus number of active stations, assuming each station goes idle for a mean amount of time T_I and then active in order to transmit a message.

The best available evidence today is:

- Token passing via a ring is the least sensitive to workload, and offers short delay under light load and controlled delay under heavy load.
- Token passing via a bus has the greatest delay under light load and under heavy load cannot carry as much traffic as a ring and is quite sensitive to

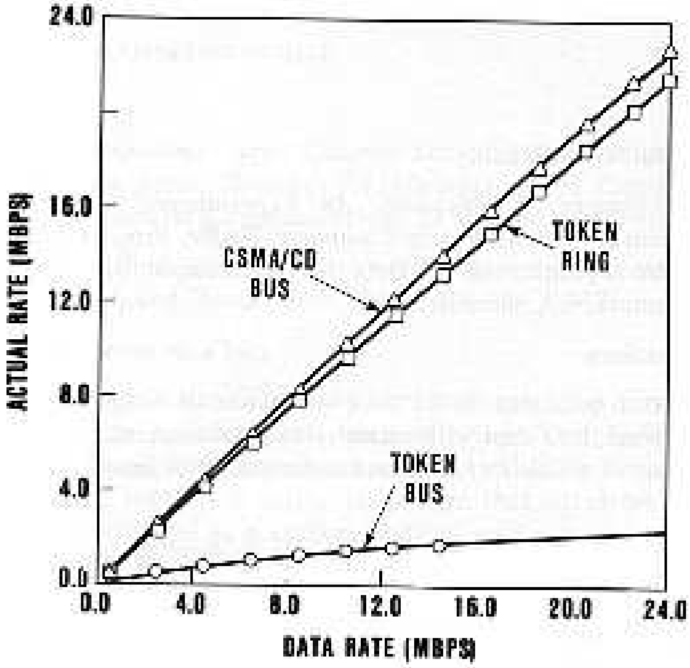


Figure 11.23. Maximum Mean Data Rate vs Clock Rate

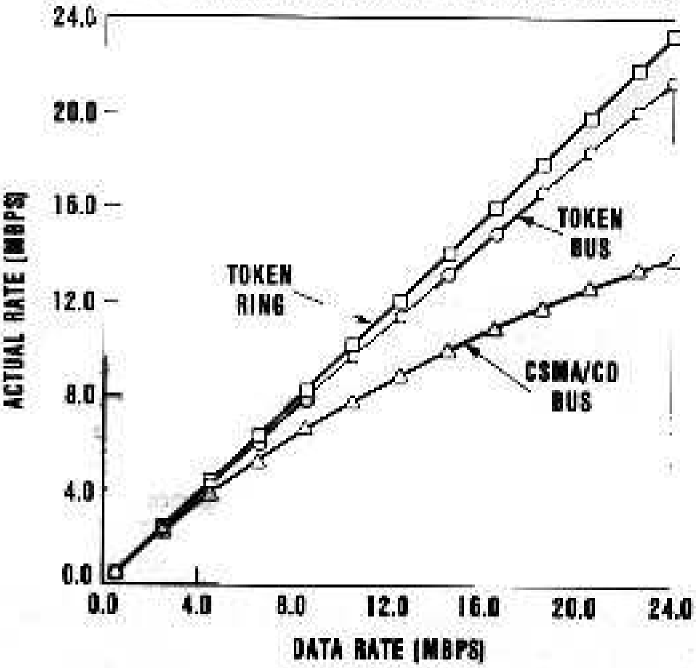


Figure 11.24. Maximum Mean Data Rate vs Clock Rate

the bus length (through the propagation time for energy to traverse the bus).

Carrier sense collision detection may offer the shortest delay under light load, or a ring may do better if the ring is sufficiently short, while it is quite

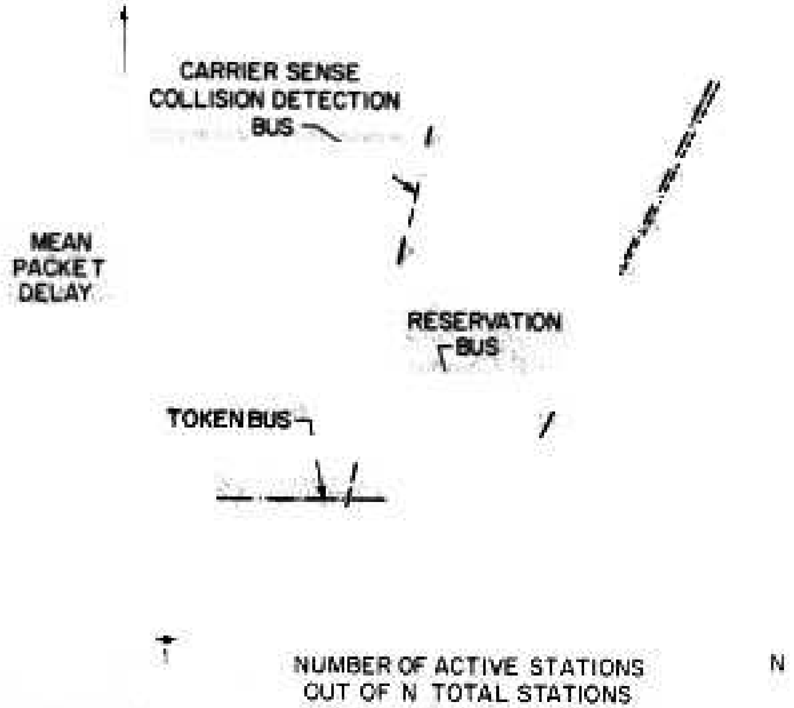


Figure 11.25.Lower Bound on Mean Message Delay vs Number of Active Stations

sensitive under heavy load to the workload and is sensitive to the bus length (the shorter the bus the better it performs) and to message length (the longer the packet the better it does).

Numerous caveats and disclaimers are in order:

- This evidence is currently being examined by those involved in actually building local area networks. No experimental data is available to confirm these plots independently. It is hoped that others will study this and subject it to independent tests.
- The scenario of one hundred stations active out of one hundred, with each generating the same workload, is possibly the *worst* congestion scenario for any system, and may in fact *never* occur in practice.

What about the impact of fluctuations, correlation, and multiple services? A great deal of work has been done in this area, and it has been summarized in earlier sections. Many argue that we should check the mean value measurements and models first, before jumping to more complicated situations. In fact, congestion analysis has had little actual impact on local area network design. However, it may have greater impact in the marketplace, where paying customers want to know how many devices of what type can be interconnected before congestion through the network becomes unacceptable. Best currently available evidence is that increasing the local area network bandwidth offers an excellent avenue for reducing transmission medium congestion. A common theme heard again and again is that it is probably better to deploy these systems and see how people really wish to use them, gathering and analyzing traffic measurements, before doing anything else.

11.12.1 Additional Reading

[1] E.Arthurs, B.W.Stuck, W.Bux, M.Marathe, W.Hawe, T.Phinney, R.Rosenthal, V.Tarassov, *IEEE Project 802 Local Area Networks Standards Traffic Handling Characteristics Committee Report Working Draft*, 1 June 1982, IEEE Computer Society, Silver Spring, Maryland.

[2] D.D.Clark, K.T.Pogran, D.P.Reed, *An Introduction to Local Area Networks*, Proceedings IEEE, **66** (11), 1497-1517 (1978).

[3] J.F.Hayes, *Local Distribution in Computer Communications*, IEEE Communications Magazine, **19** (2), 6-14(1981).

[4] IEEE Computer Society, *IEEE Project 802 Local Area Network Standards, Draft D 802.2 (Logical Link Control), Revision D 802.3 (CSMA/CD Access Method and Physical Layer Specification), Draft D 802.4 (Token-Passing Bus Access Method and Physical Layer Specification)* IEEE Computer Society, Silver Spring, Maryland (1983).

Problems

1) A communications system must poll two different types of sources for work. The arrival statistics for each source are independent Poisson processes, with the same mean arrival rate of 0.3 tasks per second. The service statistics for each source are independent identically distributed Erlang-2 random variables, with mean service time 1.0 seconds per task. Each source is served by a single processor as follows: the processor empties all of the work from source one in order of arrival, switches instantaneously to the other source, empties all of the work from source two in order of arrival, switches instantaneously to the other source, and so forth. Find

A. the utilization of the processor

B. the probability that a task must wait at all

C. the mean number of tasks in the system

D. the mean queueing time for a task

BONUS: Repeat the above if there is an overhead of 0. seconds to switch from one queue to the other.

BONUS: Repeat all of the above assuming *all* the work is generated at one queue versus equally balanced load

2) A multidrop communication system has multiple drops or taps from a common bus to different devices. We are given the following parameters:

- Transmission speed of 10 MBPS
- Time for energy to propagate from end of the bus to other worst case 20 μsec
- Time for each station to detect carrier and do polling protocol processing 100 μsec
- Fraction of time bus is busy transmitting data bits is 0.5
- Fifty stations connected to the communication system

We wish to examine two cases:

[1] All the messages are generated by one station

[2] Each station generates the same workload

For each of these cases

A Calculate the mean message delay for each of the above cases, assuming constant, Erlang-2, and exponential message length distributions.

B Repeat the above if the number of stations is changed to twenty stations and to one hundred stations.

C Repeat all of the above if the utilization changes to 0.9.

3) A four story office building has two corridors per floor, with two rows of offices per corridor. Each office is fifteen feet by fifteen feet square. Each office has two local area network sockets on each of the two walls perpendicular to the wall with the door to the corridor. Each office has a nine foot ceiling with a false hung ceiling one foot below the actual ceiling. Each floor has a wiring closet at the northeast corner. Assuming wiring or cable or optical fiber can be run only in the walls and ceiling, estimate the number of feet of cabling required for

A. A star configuration for each floor connecting each office to the closet via three pairs of wire, and each closet connected by a vertical riser of coaxial cable.

B. A coaxial cable bus with a drop from the ceiling to each outlet of four pairs of twisted wires.

C. A fiber optic ring that connects each outlet to the wiring closet, and each closet is connected by a vertical fiber optic riser.

4) A coaxial cable is used in a local area network. Electromagnetic energy propagates through the cable at a speed of 200 meters/μsec. The bus is one kilometer long. The clock transmission rate is ten million bits per second. Each station transmits a 1000 bit packet, which includes both control bits and data bits.

A. If the cable is used as a token passing bus, with the electronics at each station requiring 5 μsec per token pass, how long does it take to pass the token through one hundred stations, if no station has a packet to transmit? if every station has a packet to transmit?

B. If the cable is used as a token passing ring, with the electronics delay at each station requiring one bit transmission time per token pass, how long does it take to pass the token through one hundred stations, if no station has a packet to transmit? if every station has a packet to transmit?

C. Repeat if the clock transmission rate is one hundred million bits per second.

D. Repeat all the above if the packet size is changed to 10,000 bits per packet.

E. Repeat all the above if the bus is ten kilometers long.

5) Four stations are interconnected via a token passing bus. The mean arrival rate of messages to each station is denoted by $\lambda_K, K=1,2,3,4$. Stations one, two and three transmit a one thousand bit frame if they have any message to send; station four transmits a four thousand bit frame if it has a message to send. The token passing bus clock rate is one million bits per second. The time to pass the token from one station to another is ten microseconds.

A. If each station is visited only once during a token passing polling cycle, and each station transmits until it empties its buffers, what is the admissible region of mean throughput rates?

B. If each station is visited only once during a token passing polling cycle, and each station transmits up to one thousand bits per visit, what is the admissible region of mean throughput rates?

C. If stations one, two and three are visited only once during a token passing polling cycle, while station four is visited four times during a token passing polling cycle, and each station can transmit up to two thousand bits per visit, what is the admissible region of mean throughput rates? Repeat if each station can transmit up to one thousand bits per visit. Repeat if station four is visited two times during a token passing polling cycle.

D. Repeat if the token passing overhead is zero, or is twenty microseconds.

6) A carrier sense collision detection local area network bus is operated as a loss system: if carrier is absent, a station attempts to transmit, and is either successful or garbles its transmission with that of one or more other stations. Transmission attempts are generated according to Poisson statistics with mean arrival rate λ. Each message has a constant length of one thousand bits. The bus clock rate is one million bits per second.

A. What is the largest arrival rate that results in no more than one in one million packets undergoing a collision? No more than one in one thousand packets undergoing a collision? No more than one in ten packets undergoing a collision?

B. Repeat if the packet mean length is fixed at one thousand bits, but the packet length statistics fit an exponential distribution.

C. Repeat if the bus clock rate is ten million bits per second.

7) N stations are connected to a local area network bus. Each station generates packets with a mean length of one thousand bits. The local area network bus clock rate is one million bits per second. Electromagnetic energy can propagate from one end of the bus to the other worst case in 20 μ seconds, which we call a slot time, denoted T_{slot}. The bus access method is slotted P-persistent collision detection: If the previous time slot was idle, a ready station begins to transmit. If the station successfully seizes the transmission medium for one slot time, it continues to transmit the message to completion. If the station is not successful in seizing the transmission for one slot time, it ceases transmission and will retry in the next time slot with propability P and otherwise defer to the next slot time and repeat the decision process all over again.

A. What is the maximum data transmission rate versus the bus clock rate?

B. Suppose all but one station always have a message to transmit, and the final station generates messages according to Poisson statistics with mean arrival rate λ. What is the largest value of λ such that the mean packet queueing time (both transmission and waiting) is under ten milliseconds for twenty stations total? for one hundred stations total? Repeat if the bus clock rate is ten million bits per second. Repeat if the bus clock rate is one hundred million bits per second.

8) Eight stations are connected to a local area network bus. The bus can be in one of three states: idle, collision, and successful message transmission. The mean time in the idle state and collision state are equal to 20 μ seconds. The mean time in the successful message transmission state equals the time to transmit a one thousand bit packet. A binary decision tree is used to determine station retry priorities.

A. The bus clock rate is one million bits per second. If the station addresses are encoded into three bits, what is the worst case and best case time to the first successful packet transmission? What is the worst case and best case time to allow every station to transmit one packet?

B. Repeat if the bus clock rate is ten million bits per second.

C. Repeat if the bus clock rate is one hundred million bits per second. Plot the maximum mean data rate versus the clock rate.

D. Repeat if the number of stations is increased to one hundred and twenty eight.

E. Repeat for the case where the total number of packets transmitted per frame is fixed, but instead of each station generating a packet, only one station generates all the packets.

9) A local area network bus interconnects one hundred stations. Each station is either idle for a mean time of 10 milliseconds, or active waiting to transmit or transmitting a constant length 1000 data bit packet. Each packet also has one hundred control bits associated with addressing, sequencing, and error detection. The bus transmission rate is ten million bits per second. The bus can be accessed in three different ways:

[1] Token passing: each station passes the token to the next station with an overhead time of 10 μ seconds for propagation plus five microseconds per station (to allow electronics to quiesce).

[2] Carrier sense collision detection: each station waits for the transmission medium to be idle, and then transmits; if the transmission is successful within a slot time equal to two one way propagation times (20 microseconds) plus five microseconds per station (to allow electronics to quiesce) then the transmission continues, and otherwise it is aborted and will be attempted at a later point in time. The station enters back into its idle state and the process starts anew.

[3] Decision tree: each station waits for the start of a variable length frame, and then transmits. If the transmission is successful within a slot time equal to two one way propagation times (20 microseconds) plus five microseconds per station (to allow electronics to quiesce), then the transmission continues, and otherwise it is aborted and is retried according to the station binary address (lower addresses have higher priority).

Answer the following questions:

A. Plot a lower bound on mean delay versus number of stations for each access method.

B. Repeat if the bus transmission speed is ten million bits per second. Repeat if the bus transmission speed is one hundred million bits per second.