A Theoretical Performance Analysis of Polling and Carrier Sense Collision Detection Communication Systems

E.Arthurs
B.W.Stuck

Bell Laboratories
Murray Hill, New Jersey 07974

ABSTRACT

A set of stations wish to transmit messages to one another over a shared link. Messages are generated either according to so called *finite source* or *infinite source* arrival statistics. The message lengths are independent identically distributed random variables. Two policies for arbitrating contention are compared: *polling*, and *carrier sense collision detection*. The maximum mean throughput rate and mean message delay is calculated as a function of model parameters.

1. Introduction

At present there is interest in determining fundamental limitations on the traffic handling characteristics of several proposed media access methods for so called *local area networks*. (cf. Kryskow and Miller, 1981). This report is a summary of a study to address some of these issues.

We fix as input parameters the arrival statistics, the message length statistics, and the arbitration, and the physical topology, and examine as outputs the mean throughput rate and mean delay of the transmission system. We caution that many other factors must be considered in designing a local network; this report only addresses one aspect of such a system.

2. Summary

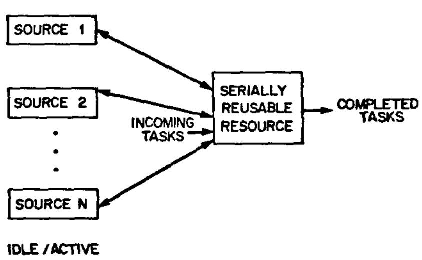
The principal findings are

- An admissible region of mean delay and mean throughput rate for polling or token passing arbitration policies; at present, a similiar set is unknown for carrier sense collision detection policies.
- A new bound on mean delay is calculated for special cases of carrier sense collision detection bus and token passing polling arbitration policies for busses and rings

We strongly stress that many factors besides traffic handling characteristics of the transmission medium must be considered in assessing one design choice against alternatives.

3. Problem Statement

The figure below shows a block diagram of the system under consideration.



In order to specify the mathematical abstraction of a real situation we must describe

- arrival statistics for each transaction type
- processing time statistics for each type of transaction
- arbitration or scheduling policy when multiple transactions are present for the serially reusable resource

In addition, the physical topology must be specified; here we deal with two different topologies, a *bus* where every station can transmit to every other station over a common transmission medium, and a *ring* where each station receives from only one station and transmits to only one station.

Time can be either *discrete* (time is marked with so called *slots)* or *continuous* (the time scales of interest are much greater than any basic granularity due to clocking and synchronization). Throughout most of this report we adopt *continuous* time descriptions of system behavior.

The arrival statistics for the so called *finite source* model of each station are as follows:

- each of N stations are either idle or active
- the idle times for each station form a sequence of independent identically distributed exponential time intervals

The arrival statistics for the so called *infinite source* model of each station are as follows:

- the interarrival times between messages for each station form a sequence of independent identically distributed exponential random variables

The principal difference is that the station cannot transmit any more messages until it returns to the idle state in the finite source model, while the infinite source model does not have this restriction. Both models will be of use: the finite source model in many cases can quite closely approximate first order behavior of a station, but the resulting analytic complexities lead us to analytically tractable models such as the infinite source model. It is gratuitous to claim that one model is better than another in a given situation unless one compares measurements with analysis. If we start with a set of N sources with mean idle time *TIDLE* and allow both N and *TIDLE* to become infinite while their ratio is fixed, then the resulting arrival stream obeys infinite source arrival statisitcs.

The message lengths can be statistically characterized as follows:

- the active transmission times for each source form a sequence of independent identically distributed time intervals drawn from an arbitrary distribution

Each station is assumed to have an infinite capacity buffer for storing messages from this point on. The scheduling or arbitration policies considered here are as follows:

CH1694-9/81/0000/0156$00.75

- *Polling*– Messages are transmitted in a cyclical sequence, by polling stations in a fixed order with the option of visiting a station one or more times during a polling cycle and removing up to a fixed maximum number of messages from the buffer associated with each station for holding messages; polling stations in a fixed order until a message is transmitted and then returning to poll in a fixed order is called *static priority* arbitration, because each station is assigned a priority or urgency based on the order in which it is polled; polling stations in a cyclic order until a message is transmitted and then polling the next station on the list is called polling from this point on; once message transmission is initiated it is not interrupted, i.e., we confine attention to *nonpreemptive* arbitration policies
- *Carrier sense collision detection*-- Each station senses the state of the transmission medium; if it is idle, and a station has a message, it begins transmission, and either succeeds or contends with one or more other stations that are also attempting to transmit; a variety of actions are possible if contention is detected, such as assigning stations fixed or randomized time intervals following contention detection to begin retransmission

4. Asymptotic Analysis for Noncollision Polling Policies

In order to gain insight, we examine limiting cases of the traffic handling characteristics of noncollision polling arbitration policies. No parallel analysis is known at present for carrier sense collision detection policies.

4.1 One Message 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. We examine this case both with and without overhead incurred in switching from one source to another.

We denote by $THROUGHPUT(K)$ the mean throughput rate of transactions from source $K, K=1,...,N$. We denote by $TMESS(K)$ 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. Each source is visited $VISIT(K)$ times per polling cycle. For a variety of reasons such as load balancing, it is often desirable to visit a given source more than once in a polling cycle.

Using *Little's result*, we assert that 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:

$$UTIL(K) = THROUGHPUT(K) \times TMESS(K)$$

where $UTIL(K)$ 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:

$$\sum_{K=1}^{N} UTIL(K) = \sum_{K=1}^{N} THROUGHPUT(K) \times TMESS(K) \leq 1$$

To include the effects of overhead in a polling or cyclic scheduling policy, a slightly different treatment must be used. During a polling cycle, the resource 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 TOV 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. We denote by $POLLING\ CYCLE$ the duration of one complete polling period. Since our assumption states that the system is busy either doing useful work or doing overhead, the polling cycle must satisfy

$$POLLING\ CYCLE = TOV + \sum_{K=1}^{N} [TMESS(K) \times VISIT(K)]$$

Since the assumption states that each source always has a message to be sent,

$$THROUGHPUT(K) = \frac{VISIT(K)}{POLLING\ CYCLE}$$

4.2 Polling

Assume the system is always busy with polling overhead or message transmission, but the sources may be busy or idle. 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.

$$\frac{TOV}{POLLING\ CYCLE} + \sum_{K=1}^{N} THROUGHPUT(K) \times TMESS(K) = 1$$

Solving for $POLLING\ CYCLE$, we get

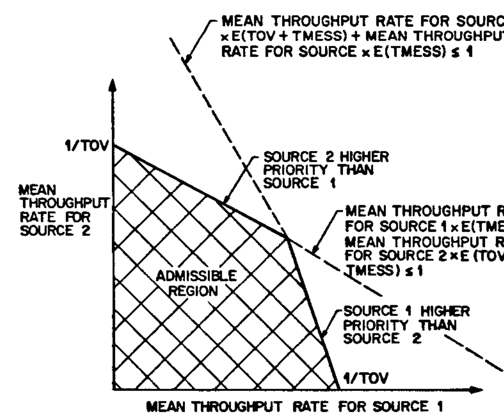
$$POLLING\ CYCLE = \frac{TOV}{1 - \sum_{K=1}^{N} THROUGHPUT(K) \times TMESS(K)}$$

Note that the duration of a polling cycle is proportional to TOV at all traffic rates! 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,

$$THROUGHPUT(J) \leq \frac{VISIT(J)}{POLLING\ CYCLE} \quad J=1,...,N$$

$$THROUGHPUT(J) \leq VISIT(J) \times \frac{1 - \sum_{K=1}^{N} THROUGHPUT(K) \times TMESS(K)}{TOV}$$

The figure below plots an illustrative admissible region of mean throughput rates for the special case of two sources.



AN ILLUSTRATIVE EXAMPLE

TWO SOURCES, ONE VISIT EACH PER POLLING CYCLE

This is a convex set, with the convex hull given by the static priority policies.

If the sources have identical traffic characteristics, then the mean throughput rate and mean message length are independent of the station index, so

$$THROUGHPUT \leq \frac{1 - N \times THROUGHPUT \times TMESS}{TOV}$$

$$THROUGHPUT \leq \frac{1}{TOV} + N \times TMESS$$

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 $TDELAY(K)$, or is idle for a mean time interval $TIDLE(K)$. The mean throughput rate for each source, by definition, is simply the reciprocal of the sum of these two time intervals:

$$THROUGHPUT(K) = \frac{1}{TDELAY(K) + TIDLE(K)} \qquad K=1,...,N$$

If we rearrange this, we see that

$$TDELAY(K) = \frac{1}{THROUGHPUT(K)} - TIDLE(K) \qquad K=1,...,N$$

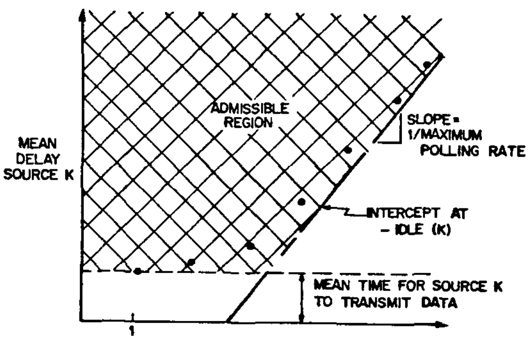
Using the earlier upper bound on throughput for identical sources,

$$TDELAY(K) \geq TOV + (N \times TMESS) - TIDLE$$

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

$$TDELAY \geq C0 + [N \times (C1+TMESS)] - TIDLE$$

The figure below shows an illustrative delay characteristic for a given source.



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!
- Heavy loading--the serially reusable resource is heavily loaded, so the 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 $TDELAY(K)$ 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. This is the subject of a later section.

4.3 Finite Source Mean Throughput and Mean Delay Asymptotes

What if we have one terminal or souce on the system, i.e., N=1? Then we see

$$TOV + TMESS + TIDLE = POLLING\ RATE \qquad \textit{one terminal}$$

assuming there is always a message or task to be processed on each visit, and thus

$$THROUGHPUT = \frac{1}{TOV + TMESS + TIDLE} \qquad \textit{one terminal}$$

If N is greater than one, and each source has a task on each visit, then

$$N\times TOV + N\times TMESS + TIDLE = POLLING\ CYCLE \qquad \textit{N terminals}$$

$$\rightarrow THROUGHPUT = \frac{N}{N\times TOV + N\times TMESS + TIDLE} \qquad \textit{N terminals}$$

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

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

$$\frac{UTIL}{TMESS} = \frac{N}{TIDLE + TDELAY}$$

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

$$\lim_{N\to\infty} \frac{UTIL}{TMESS} = \frac{UMAX}{TMESS}$$

where $UMAX$ is the maximum utilization of the serially reusable resource. Different arbitration policies result in different values of $UMAX$. For example, for the case of zero overhead or time required to resolve contention, the maximum utilization is simply one, while for polling and carrier sense collision detection it is less than one.

5. Symmetric Stations Using Token Passing Arbitration

The operation of a polling system may be thought of as passing a token from station to station, with the owner of the token possessing control of the transmission medium, i.e., the token possessor is the highest priority transmitter, and priorities change with time as the token is passed. We summarize analysis of two different mathematical abstractions of this system, one involving a set of N stations each of which is either idle (no message) or active (waiting to transmit or transmitting a message), and one where messages are generated at N stations according to simple Poisson arrival statistics. The first model may be thought of as having a buffer at each station that can only hold one message at most, while the second model involves infinite buffering of arbitrarily many messages. In the queueing literature, the first model of arrival statistics is called the *finite source* model while the second model is called the *infinite source* model that results from allowing the number of finite sources to become infinite while the mean idle time per source also goes to infinite but the *ratio* of the number of sources divided by the mean idle time is constant. This work builds and extends work by many others over the past two decades some of whom we cite here (Avi-Itzhak, Maxwell, Miller, 1963; Boudreau, Griffin, Kac, 1962; Cooper, 1970; Eisenberg, 1972; Hashida, Kawashima, 1979; Rubin, DeMoraes, 1981; Spaniol, 1979; Sykes, 1970).

5.1 Finite Source Model

- N identical stations attempt to transmit messages; each station is either idle or is active (either waiting to transmit or transmitting). The idle times for all stations form a sequence of independent identically distributed exponential random variables with mean idle time $1/\lambda$
- the message lengths transmitted by each station form a sequence of independent identically distributed random variables, with associated transmission time $TMESS$ which has an associated moment generating function $\gamma_{TMESS}(z)$ defined by

$$E[exp(-zTMESS)] = \gamma_{TMESS}(z)$$

- the policy for operating the communications link is as follows: each station is polled according to a cyclic order, and when its turn in the polling sequence arrives each station transmits a message if it has one or passes control onto the next station in the polling sequence. The time to pass control on to the next station is a random variable; in our analysis we will be concerned with the distribution of the random variable denoting the time for all stations to be polled once assuming *no* station has any message to transmit. This random variable, denoted by *TOV*, is assumed to be drawn independently from an arbitrary distribution with moment generating function $\gamma_{TOV}(z)$ defined by

$$E[\exp(-zTOV)] = \gamma_{TOV}(z)$$

In practice *TOV* may equal a constant plus a term that is linearly proportional to the number of stations on the link, for example.

The mean rate of completing messages is simply the fraction of time the link is actively transmitting messages divided by the mean time to transmit one message. We denote by *UTIL* the link utilization, while $E(TMESS)$ is the mean message transmission time, and see

$$\textit{mean throughput rate} = \frac{UTIL}{E(TMESS)}$$

Since each station is either idle or active, the mean cycle time for one station to go from idle to active and back to idle is simply

$$\textit{mean station cycle time} = E(TIDLE) + E(TDELAY)$$

and by definition the mean throughput rate is the number of stations divided by the mean cycle time per station:

$$\textit{mean throughput rate} = \frac{N}{E(TIDLE) + E(TDELAY)}$$

Equating these two expressions, we see

$$E(TDELAY) = \frac{N\ E(TMESS)}{UTIL} - E(TIDLE)$$

We omit in the interest of brevity the derivation that the mean utilization is given by

$$UTIL = \frac{FRACT\ E(TMESS)}{E(TOV)/N + FRACT\ E(TMESS)}$$

where

$$FRACT = \frac{\sum_{J=0}^{N-1} \binom{N-1}{J} \prod_{K=0}^{J} [\exp(\lambda(TOV + K\ E(TMESS)))-1]}{1 + \sum_{J=1}^{N} \binom{N}{J} \prod_{K=0}^{J-1} [\exp(\lambda(TOV + K\ E(TMESS)))-1]}$$

For the limiting special case of negligible overhead compared to the service time per job, we see that the mean flow time for cyclic polling matches that for service in order of arrival (Jaiswal, 1968): in both cases we are computing the Laplace Stieltjes transform of the service time distribution at evenly spaced points, as expected. This corrects an oversight in the literature (Hashida, Kawashima, 1979).

5.2 Infinite Source Model with Exhaustive Service

- N stations each generate messages according to simple Poisson arrival statistics, with mean message interarrival time $1/\lambda$
- The messages form a sequence of independent identically distributed random variables drawn from an arbitrary distribution with message transmission time denoted by *TMESS*
- 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 *TOV* passes doing overhead work before reaching the next station; all messages are removed from that next station, included 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

For N stations, the fraction of time the system is not transmitting messages, averaged over a suitably long time interval, is simply the ratio of the mean overhead time to go from station to station, divided by the total mean cycle time, the time interval from the epoch when a station is first visited during a cycle till the time of its next visit:

$$1 - N\ UTIL = \frac{E(TOV)}{E(C)} \rightarrow E(C) = \frac{E(TOV)}{1 - UTIL}$$

provided we can keep up with the work, i.e., provided $N\ UTIL < 1$. If we substitute this into the previous expressions, we find

$$E(TWAIT) = \tfrac{1}{2}E(C)\frac{1 - UTIL}{1 - N\ UTIL} + \frac{VAR(C)}{2E(C)}[1 - UTIL]$$

$$E(TWAIT) = \frac{N\lambda E(TMESS^2)}{2(1 - N\lambda E(TMESS)} + \tfrac{1}{2}E(TOV)\frac{1 - \lambda E(TMESS)}{1 - N\lambda E(TMESS)}$$

In the limit as the overhead becomes negligible, we see

$$\lim_{E(TOV)\to 0} E(TWAIT) = \frac{N\lambda E(TMESS^2)}{2(1 - N\lambda E(TMESS)}$$

which is the same mean time to wait as for service in order of arrival, as expected. This gives us a *lower* bound on the mean waiting time with polling:

$$E_{polling}(TWAIT) \geq E_{order\ of\ arrival}(W)$$

The second term in the mean waiting time expression using polling is due to irregularities or fluctuations in the cycle times.

The mean delay, queueing plus transmission, is given by

$$E(TDELAY) = E(TWAIT) + E(TMESS)$$

It is straightforward to extend this to encompass

- batch or compound Poisson arrival streams, where the interarrival times of batch sizes are independent identically distributed exponential random variables, and at each arrival epoch we sample independently from a common discrete distribution the number of messages arriving, to allow for phenomena such as control packets followed by data packets
- multiple visits to the same station during a polling cycle, to allow load balancing: more heavily loaded stations are visited more frequently (in a subsequent section we examine the other possibility for load balancing: only removing a finite number of messages on each visit, with the number removed being proportional to the load)
- Different amounts of overhead in moving from station to station, to encompass a variety of higher level protocol activity that might require different amounts of processing time depending upon the type of station

5.3 Infinite Source Model for One Station with Nonexhaustive Service

In order to assess the sensitivity of our results due to underlying assumptions, we consider a model of operation as seen from one station. An analysis of the multiple station case is apparently open at the present time, but the best available evidence suggests that the insight gained in the single station case can be applied to the multiple station problem.

- the arrival statistics to the queue are simple Poisson, i.e., the interarrival times are independent identically distributed exponential random variables with mean interarrival time $1/\lambda$ (the extension to batch Poisson is straightforward and is omitted)

- the service required of each task is random, with the service times forming a sequence of independent identically distributed random variables drawn from an arbitrary distribution with moment generating function $\gamma_{TMESS}(z)$
- the processor serves up to a maximum of S tasks at the buffer and then leaves for an intervisit time interval, with the intervisit random variable denoted V, and the sequence of intervisit times being independent identically distributed random variables drawn from a distribution with moment generating function $\gamma_V(z)$
- all tasks are served in order of arrival, first come, first served
- the tasks are stored in an infinite capacity buffer

This model captures the fluctuations in the cycle time distribution, but not the correlation in polling cycles due to surges of work at a particular queue.

In order for a nontrivial equilibrium distribution to exist, we demand that

$$\lambda\left[\frac{SE(TMESS)+E(V)}{S}\right] < 1$$

must be satisfied. Note that for $E(V)=0$ this is simply that the mean arrival rate must be less than the mean service rate.

The method of analysis draws on techniques pioneered in fluctuations of sums of random variables (e.g., Feller, 1966). An alternate approach to this problem is algorithmic in nature (Neuts, 1979) but is not pursued here in the interest of brevity.

5.3.1 Maximum Number of Messages Transmitted per Visit=1(S=1) The simplest case is where the processor does one task and then leaves for an arbitrary time interval (to do work elsewhere). This lengthens the service required for each task. The moment generating function of the long term time averaged waiting time distribution is given by

$$E[exp(-zTWAIT)] = \frac{[1-\lambda(E(TMESS)+E(V))]z}{z-\lambda[1-\gamma_{TMESS}(z)\gamma_V(z)]}\,\frac{1-E[exp(-zV)]}{zE(V)}$$

where the second term is due to the fact that a task might arrive during the absence of the processor and thus must wait for it to return. The flow time is simply the sum of the waiting time plus the service time for a task, and hence the moment generating function of the long term time averaged flow time distribution is given by

$$E[exp(-zTDELAY)] = E[exp(-zTMESS)]\times E[exp(-zTWAIT)]$$

The mean waiting time is given by

$$E(TWAIT) = \frac{\lambda E[(V+TMESS)^2]}{2(1-\lambda[E(V)+E(TMESS)])}$$

while the mean flow time is

$$E(TDELAY) = E(TWAIT) + E(TMESS)$$

The moment generating function of the number in system process is given by

$$E[x^N] = E[exp(-\lambda(1-x)TDELAY)]$$

and hence the mean number in system is

$$E(N) = \lambda\, E(TDELAY)$$

5.3.2 Maximum Number of Messages Transmitted per Visit=2(S=2) The extension of these results from $S=2$ to $S>2$ is straightforward once the basic ideas developed here for this special case have been mastered, and thus we concentrate on a moderately thorough discussion of this case.

We consider a sequence of independent random variables $\{X_k\}$ where

$$E[exp(-zX_k)] = \begin{cases}\gamma_{TMESS}(z)\gamma_V(z)\frac{\lambda}{\lambda-z} & k\ even\\ \gamma_{TMESS}(z)\frac{\lambda}{\lambda-z} & k\ odd\end{cases}$$

We also define a random walk with partial sums denoted by S_k where

$$S_k = S_{k-1} + X_k \quad k\geq 1$$

$$E[exp(-zS_0)] = \frac{\lambda}{\lambda-z}\,\frac{1-\gamma_V(z)}{1-\gamma_V(\lambda)}$$

We define a new random variable T as the smallest value of k such that $S_k<0$. Finally, we define $\phi_i(z)$ as

$$\phi_i(z) = E[exp(-zS_i)\ ;\ T>i]$$

We do so because the long term time averaged waiting time distribution has moment generating function given by

$$\lim_{n\to\infty} E[exp(-zTWAIT_n)] = \frac{\Phi(z)}{\Phi(1)}$$

$$\Phi(z) = \sum_{i=0}^{\infty}\phi_i(z)$$

It is useful to define an auxiliary generating function

$$\Theta_i(z) = E[exp(-zS_i)\ ;\ T=i]$$

which obeys the following recursions:

$$\phi_i(z)+\Theta_i(z) = \phi_{i-1}(z)\gamma_{TMESS}(z) \quad i\ odd$$

$$\phi_i(z) + \Theta_i(z) = \phi_{i-1}(z)\gamma_{TMESS}(z)\gamma_V(z) \quad i\ even$$

We can rewrite these recursions as follows:

$$\sum_{i=0}^{\infty}\phi_{2i+1}+\frac{\lambda}{\lambda-z}\sum_{i=0}^{\infty}PROB[T=2i+1] = \frac{\lambda\gamma_{TMESS}(z)}{\lambda-z}\sum_{i=0}^{\infty}\phi_{2i}(z)$$

and

$$\phi_0(z) + \sum_{i=1}^{\infty}\phi_{2i}(z) + \frac{\lambda}{\lambda-z}\sum_{i=1}^{\infty}PROB[T=2i] =$$

$$\frac{\lambda\gamma_{TMESS}(z)\gamma_V(z)}{\lambda-z}\sum_{i=0}^{\infty}\phi_{2i+1}(z) + \frac{\lambda\gamma_V(z)}{\lambda-z}\sum_{i=0}^{\infty}\phi_{2i+1}(z) + \frac{\lambda}{\lambda-z}\,\frac{1-\gamma_V(z)}{1-\gamma_V(\lambda)}$$

If we solve for $\Phi(z)$ we find that

$$\Phi(z) = \frac{NUMERATOR}{(1-\gamma_V(\lambda))[(\lambda-z)^2-\lambda^2\gamma_p^2(z)\gamma_V(z)}$$

where $NUMERATOR$ is given by

$$NUMERATOR = -\lambda(\lambda-z)[\gamma_V(z)-\gamma_V(\lambda)]+\lambda^2\gamma_{TMESS}(z)[1-\gamma_V(z)]-[1-\gamma_V(\lambda)]\times$$
$$\lambda^2\gamma_{TMESS}(z)[\gamma_V(z)PROB[T\ odd] + PROB[T\ even]]$$

where we have used the obvious notation for the probability of the events that T is even and T is odd respectively.

We note that the denominator is zero for some real value of $z, Re(z)>0$ but since $\Phi(z)$ is a moment generating function, it is an analytic function for $z>0$ and hence the numerator must also be zero at the same point. Thus, we have two conditions that must be satisfied:

- Cancellation of the zero in the denominator with the zero in the numerator
- Normalization, i.e., the moment generating function evaluated at z=0 must be unity

Satisfying both conditions is equivalent to evaluating the unknown constants for the probability of the event T is even and the event T is odd. While this is numerically tractable, little analytic insight is available at present into the formulae, even after considerable algebraic manipulations.

What is the analysis required for $S>2$? First, $S-1$ zeroes in the denominator that must be cancelled by the same zeroes in the numerator. The (S-1) constants involve evaluating the probability of the event $T=k$ for $k=0,1,...,S-2$ which will be implicitly evaluated by the (S-1) zero cancellations plus the normalization condition. We leave this as an exercise.

The flow time is the sum of the waiting time plus service time, and hence the moment generating function of the long term time averaged flow time distribution is given by

$$E[exp(-zTDELAY)] = E[exp(-zTMESS)]\, E[exp(-zTWAIT)]$$

Using a generalization of Little's law, we find that the moment generating function of the long term time averaged distribution of number in system is given by

$$E[x^N] = E[\exp(-\lambda(1-x)TDELAY)]$$

There are a variety of numerical methods for approximating the waiting and flow time distributions as well as their associated moments. At the present time little insight is available into the probabilistic significance of the (S-1) roots or how they relate back to the original model parameters.

In a pioneering paper, Boudreau et al (1962) showed that if the service time is zero then as the utilization of the serially reusable resource approaches unity, the roots obey an *equidistribution* type theorem, i.e., the roots asymptotically are uniformly spaced about the unit disk. Moreover, they showed that the mean waiting time and mean number in system are a simple function of these roots. For the case where the intervisit time is constant and of duration T while the mean arrival rate is λ they showed that the flow time is given by

$$E(TDELAY) = \frac{T}{2(S-\lambda T)} + \frac{T}{2} - \frac{S-1}{2\lambda} + \frac{1}{\lambda}\sum_{K=1}^{S-1}\frac{1}{1-ROOT(K)}$$

which makes it clear how important the S-1 roots are in determining system performance.

5.3.3 Exhaustive Service $(S=\infty)$ Given all the previous assumptions, the moment generating function for the long term time averaged waiting time distribution is given by

$$E[exp(-zTWAIT)] = \frac{[1-\lambda E(TMESS)]z}{z-\lambda[1-E(exp(-zTMESS))]}\;\frac{1-E[exp(-zV)]}{zE(V)}$$

This random variable has a mean of

$$E(TWAIT) = \frac{\lambda E(TMESS^2)}{2(1-\lambda E(TMESS))} + \frac{E(V^2)}{2E(V)}$$

The flow time, the time from arrival until departure, is denoted by the random variable *TDELAY* with associated moment generating function

$$E[exp(-zTDELAY)] = E[exp(-zTMESS)] \times E[exp(-zTWAIT)]$$

$$E(TDELAY) = E(TWAIT) + E(TMESS)$$

The moment generating function for the number in the system at arrival epochs, completion epochs, or arbitrary time epochs, is given by

$$E(x^N) = E(exp(-(\lambda(1-x)TDELAY)))$$

$$E(N) = \lambda E(TDELAY)$$

6. Carrier Sense Collision Detection Contention Bus Arbitration

We now sketch analyses of the traffic handling characteristics of variations on carrier sense collision detection bus arbitration.

6.1 Unslotted Operation with Loss

We assume that all attempts 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;* a subsequent section will deal with slotted operation. The message transmission types are independent identically distributed random variables drawn from an arbitrary distribution denoted by $G_{TMESS}(X)$ which equals $PROB[TMESS \le X]$ with associated Laplace Stieltjes transform $\hat{G}_{TMESS}(z)$. If a source has been listening to the channel for at least *TOV* 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 is given by, from these definitions

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

$$\frac{\lambda}{\lambda+z}\left[e^{-\lambda TOV}\hat{G}_{TMESS}(z) + \int_0^{TOV}\lambda e^{-\lambda X}dXe^{-zX}\hat{G}_{\Delta}(z)E(exp(-zTSUCCESS))\right]$$

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 *TOV*. For example, the mean duration of time between successful message transmissions is

$$E(TSUCCESS) = E(TMESS) - TOV + \frac{e^{2\,TOV\,\lambda}}{\lambda}$$

The mean utilization of the channel is $E(TMESS)/E(TSUCCESS)$ which reaches its maximum when $2\lambda TOV=1$:

$$\max UTIL = \frac{E(TMESS)}{E(TMESS) + TOV(2e-1)}$$

6.2 Slotted Operation with Loss

If we parallel the above mode of operation but demand slotted operation, where time is broken up into equal length time intervals called time slots, and all attempts are made at the start of a slot, then the same analysis as above shows that

$$\max UTIL = \frac{E(TMESS)}{E(TMESS) + TOV(e-1)}$$

6.3 Unslotted Operation with Delay

The model analyzed here is as follows:

- N sources are in one of two states, idle and active. The duration of times in the idle states are independent identically distributed exponential random variables with mean $1/\lambda$. In the active state the source attempts to send a message and once it succeeds, it returns to the idle state.
- The message transmission times form a sequence of independent identically distributed random variables, drawn from a distribution with mean $E(TMESS)$.
- If a good transmission is in progress for *TOV* seconds or more, all messages arriving during this transmission defer until the completion and then all ready sources attempt to transmit; if a source encounters contention after it is active for *TOV* seconds, it retries until it successfully transmits its message; the sequence of time intervals between retries are independent identically distributed exponential random variables with mean $1/\rho$. We choose to fix ρ such that

$$N\rho = \frac{1}{2\,TOV}$$

For one station, we see

$$E(DELAY) = E(TMESS) \quad N=1$$

In general, we see

$$E(DELAY) = \frac{N\,E(TMESS)}{UTIL} - \frac{1}{\lambda}$$

and as $N \to \infty$ we see (Abramson, 1973)

$$\lim_{N \to \infty} UTIL = UTILMAX = \frac{E(TMESS)}{E(TMESS) + (2e - 1)\,TOV}$$

so that

$$E(DELAY) = N[E(TMESS) + (2e-1)\,TOV] - \frac{1}{\lambda} \quad N \to \infty$$

6.4 Slotted Operation with Delay

If we modify operation to work synchronously or with time slots, then it is straightforward to show that

$$\lim_{N \to \infty} UTIL = UTILMAX = \frac{E(TMESS)}{E(TMESS) + (e-1)TOV}$$

and hence

$$E(DELAY) = N[E(TMESS) + (e-1)TOV] - \frac{1}{\lambda}$$

In general, it is possible to use a wide variety of data structures to resolve contention within one time interval, followed by another time interval for data transmission. Examples of such data structures are trees (linked lists, balanced trees, AVL trees); here, unlike general data base problems, there is limited update and modification of the data structure (cf an airline reservation system!). For example Capetanakis (1977) (see also Tsybakov et al, 1979, 1980) has shown that used a balanced tree

$$E(DELAY) = N\left[E(TMESS) + \left(\frac{1}{0.488} - 1\right)TOV\right] - \frac{1}{\lambda}$$

References

[1] N.Abramson, *Packet Switching with Satellites*, National Computer Conference, **42** (1973).

[2] B.Avi-Itzhak, W.L. Maxwell, L.W.Miller, *Queueing with Alternating Priorities*, Operations Research, **13**, 306-318 (1963).

[3] P.E.Boudreau, J.S.Griffin, Jr., M.Kac, *An Elementary Queueing Problem*, American Mathematics Monthly, **63**, 713-724 (1962).

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

[5] M.Eisenberg, *Queues with Periodic Service and Changeover Time*, Operations Research, **20** (3) 440-451 (1972).

[6] W.Feller, **An Introduction to Probability Theory and Its Applications**, Volume II, pp.193-199, Wiley, NY, 1966.

[7] O.Hashida, K.Kawashima, *Analysis of a Polling System with Single User at Each Terminal*, Transactions of IECE of Japan, **E62**, (12), 901-903 (1979); see also **J62** B(12), 1097-1102 (1979).

[8] N.K.Jaiswal, **Priority Queues**, pp.49-51, Academic Press, NY, 1968.

[9] J.M.Kryskow, C.K.Miller, *Local Area Networks Overview--Part I: Definitions and Attributes*, Computer Design, 22-35 (February, 1981), *Local Area Networks Overview--Part II: Standards Activities*, Computer Design, 12-20 (March, 1981).

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

[11] I.Rubin, L.F.DeMoraes, *Polling Schemes for Local Communication Networks*, Proceedings International Communications Conference 1981, pp. 33.5.1-7, Denver, Colorado, 14-18 July 1981, IEEE 81CH1648-5

[12] O.Spaniol, *Modelling of Local Computer Networks*, Computer Networks, **3** (5), 315-326 (1979).

[13] J.Sykes, *Simplified Analysis of an Alternating Priority Queuing Model with Setup Times*, Operations Research, **18** (6), 1182-1192 (1970).

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