1984

Modeling a Token Ring with Non-Exhaustive Service

André B. Bondi

Report Number:

84-482

Modeling a Token Ring with Non-Exhaustive Service

André B. Bondi

Dept. of Computer Sciences

Purdue University

West Lafayette, IN 47906

CSD-TR 482

May 30th, 1984

ABSTRACT

An imbedded Markov chain analysis of a token ring with non-exhaustive service is described. The model captures the probabilistic structure of the token ring by treating the token arrivals at all stations as the imbedding points. Standard Markovian methods are used to circumvent the periodicity of the chain. The chain can be solved numerically for finite state spaces using the power method; however, the storage cost limits the use of this model to the study of very small systems. The predicted queue lengths and response times of the imbedded Markov chain analysis are compared with those of the analyses of Kuehn and Berry-Chandy. It is shown that the Berry-Chandy method necessarily underestimates the expected number of waiting packets. An improvement to their approximation is proposed.

1. Problem Description

A token ring is a local area network in which hosts are allowed to transmit data packets in rotation. In some computer systems, each packet generation causes interrupts that forces the currently executing job to relinquish the CPU until the new packet has been transmitted. In the Zurich and Pronet rings [Bux81a, Str83a, Sal82a], a single token takes a positive amount of time to scan each host in turn, stopping at those that have one or more packets awaiting transmission. In both these rings, hosts are only allowed to transmit one packet per token visit. This policy is known as *non-exhaustive service*. The term *exhaustive service* denotes a policy in which all packets present at the instant of a token's arrival may be transmitted. The exhaustive policy has been analysed by Konheim and Meister [Kon74a] Burge and Konheim [Bur71a], and by Bux [Bux81b]. Another cyclic server, a paging drum with fixed sector sizes, has been modeled by Coffman [Cof69a] and by Coffman and Denning [Cof73a]. Analyses of token cycle times in the context of machine maintenance problems have been done by Mack [Mac57a], Mack *et al* [Mac57b] and in the context of switching systems by Kuehn [Kue79a].

The non-exhaustive service discipline causes the presence of a packet at one station to increase the likelihood of there being packets at subsequent stations on the token's tour. The resulting mutual dependence of the number of packets queued for transmission at each station makes a continuous time Markovian analysis of the the ring intractable. As described by Kuehn [Kue79a], the state of the ring at time t is given by the ordered tuple

$$(N_0(t), N_1(t), \ldots, N_{k-1}(t), X(t), T(t)) \tag{1}$$

where the N's are packet queue lengths, $X(t)$ is the current position of the token, and $T(t)$ is the time the job being served has been in service (taken to be zero if there is no job in service).

The classical technique for circumventing the difficulty of a continuous time Markovian analysis is the imbedded Markov chain. In this method, the system is analysed only at time instants, often referred to as imbedding points, chosen so that the events that occur between them are stochastically independent [Kle75a, Ken53a]. The steady state distribution of the chain is then evaluated by first computing the state transition probabilities between the imbedding points and then solving the corresponding forward Chapman-Kolmogorov equations.

These equations are not the same as the global balance equations, which describe the steady state distribution in terms of state transition *rates* (see, e.g. [Cha77a]) rather than state transition *probabilities*.

In simple problems such as the M/G/1 queue, the steady state equations can be solved with generating functions [Cox65a]. In the next section, we shall see that a token ring's imbedded Markov chain of queue lengths at successive token arrival instants is periodic with degree equal to the number of hosts, so that it cannot have an equilibrium joint queue length distribution. However, the joint queue length distribution defined at the token arrival instants of a particular host does exist. As its evaluation by generating functions appears to be intractable, we shall reduce the scope of the problem by imposing an upper bound on the number of queued packets and then proceed to solve the steady state equations by numerical means. The purpose of this is two-fold. First, this analysis will provide the basis for a discussion of Kuehn's approximation method in Section 3. Second, the results of the direct computations will provide a useful benchmark for comparison with the predictions of his method and the approximate methods approximate method of Berry and Chandy [Ber83a], which will be discussed in Section 4. Fixed packet lengths and token passing overheads will be assumed throughout this paper unless otherwise stated.

2. An Imbedded Markov Chain Analysis

Consider the state of a k-station token ring at the instants at which the token arrives at a station. This may be represented by the vector

$$(N,X) \tag{2}$$

where $N = (N_0,N_1, \ldots, N_{k-1})$ is the vector of packet dispatch queue lengths at all stations, and $X=0,1,2,\ldots,k-1$ is the index of the station at which the token has just arrived. We shall make make the simplifying assumption that if the token is on the ring, no packet is currently in transit, i.e. being served. This is characteristic of the Zurich ring [Bux81a].

The equation describing the successive states is given by

$$[N_{(n)},X] = [N_{(n-1)},X-1 \ mod \ k] + [D_{n-1},0]-[H_{(X-1)}(N_{(n-1)}),0] \quad (3)$$

where $H_{(X)}(N)$ is a elementary vector with 1 in the Xth position if the Xth component of N is positive and 0 otherwise, and D_{n-1} is a (random) vector describing the number of arrivals at each station during the time the token moves to station X after arriving at station $X-1$ mod k.

Notice that this chain has a deterministic component, the station index. Also, the random vector D_{n-1} depends on the state of the system at imbedding point $n-1$, because more packets may arrive during the additional time it takes to transmit a packet than during an empty scan. It is these features of the system that make the analysis of the token ring far less tractable than that of the M/G/1 queue, whose state transition equation is similar to (3) (see, e.g. [Cox65a]). To make direct analysis of the token ring feasible, we shall restrict number of queued packets at station i to l_i-1. The state space of the imbedded Markov chain is therefore

$$[0,l_0-1]\times[0,l_1-1]\times \cdots [0,l_{k-1}-1]\times[0,k-1]$$

and has size kG, where $G=\prod_{i=0}^{k-1} l_i$. We may assume that the error caused by this restriction is small if the probability of lost packets at station i with buffer size l_i-1 is also small.

Let Q_i denote the state transition probability matrix of N between the token's arrival at station i and station $i+1$ mod k.

The state transition matrix of the whole system is given by

$$Q = diag(Q_0,Q_1,\ldots,Q_{k-1})P \quad (4)$$

where the Q_i's are all $G \times G$ matrices, and P is the generator of the group of cyclic permutation matrices of order k, containing blocks of identity matrices or 0's of the same dimensions as the Q_i's. For example, when $k=3$,

$$Q = \begin{bmatrix} O_G & Q_1 & O_G \\ O_G & O_G & Q_2 \\ Q_0 & O_G & O_G \end{bmatrix}$$

where O_G is a $G \times G$ matrix of zeroes.

If the arrival process at each station is Poisson, transition probabilities in each Q_i may be computed using standard imbedded Markov chain techniques. As mentioned after equation (3), the number of arrivals depends on whether service is rendered or not; the time during which packet arrivals may occur is either d or $d+S$, where d is the (fixed) time the token takes to move between consecutive stations, and S is the (fixed) service time of a packet. Computation of the Q_i's is simple when S is constant. For the purpose of illustration, suppose that there are two hosts numbered 0 and 1 with Poisson arrival rates λ_0, λ_1 respectively. Let $p^0_{(m_0 m_1)(n_0 n_1)}$ denote the probability that the joint queue lengths will change from (m_0, m_1) to (n_0, n_1) between token arrivals at 0 and 1 in that order. The following transitions are possible:

(a) If no packets are present at 0 when the token arrives, the joint probability of the queue lengths increasing by (m_0, m_1) is given by

$$p^0_{(0,m_1)(i,m_1+j)} = \frac{(\lambda_0 d)^i}{i!} \frac{(\lambda_1 d)^j}{j!} e^{-(\lambda_0+\lambda_1)d} , \qquad i < c_0 ,\ m_1+j < c_1 \tag{5}$$

$$= 0 \quad \text{otherwise}$$

(b) At least one packet is present at 0. The time the token takes to move from 0 to 1 is $S+d$, and one packet is removed from the queue at 0. Then,

$$p^{0}_{(m_0,m_1)(m_0-1+i,m_1+j)} = \frac{(\lambda_0(S+d))^i}{i!}\frac{(\lambda_1(S+d))^j}{j!}e^{-(\lambda_0+\lambda_1)(S+d)},$$

$$m_0+i<c_0,\ m_1+j<c_1 \qquad (6)$$

$$= 0 \quad \text{otherwise}$$

From equation (4), Q is in the canonical form of the transition matrix of a Markov chain with period k. Hence, the equation

$$p = pQ$$

has no steady state solution. However, it can be shown that Q^k is a block diagonal matrix describing the transition probabilities of N during a complete cycle of the token [Ios80a]. Furthermore,

$$p = pQ^k \qquad (7)$$

does have a steady state solution. The first entry of Q^k is $Q_0Q_1 \ldots Q_{k-1}$; the kth entry is $Q_{k-1}Q_0Q_1 \ldots Q_{k-2}$. Each matrix product in Q^k represents a self-contained irreducible aperiodic Markov chain describing the state of the system when the token arrives at the corresponding station. Because closed form expressions or generating functions for the steady state distribution of these chains are not not readily available, the distributions must be obtained by numerical means.

The joint steady state queue length probabilities q_i at each station i may be computed by applying the power method [Dah74a] to each diagonal block of Q^k in turn For example, q_0 may be found by repeatedly performing the iteration

$$q_0^{(l)} = q_0^{(l-1)}Q_0Q_1 \ldots Q_{k-1} \qquad (8)$$

where $q_0^{(0)}$ is chosen so that its components sum to one. If the packet arrival rates at all hosts are the same, i.e. the system is symmetric, this need only be done for one block.

The computational cost of this method is high. The structure of each of the Q_i's is such that the storage of pointers for sparse matrix multiplication methods exceeds the cost of storing zeroes. Once the Q_i's have been multiplied, the cost of each iteration is $O(G^2)$, and G grows exponentially with the number of hosts. The cost of solving a model with a heavy load is especially high, (a) because G must be very large to achieve a reasonable level of accuracy and (b) because the iterative scheme (8) converges slowly, since the Markov chain it describes takes longer to reach steady state than one with a light load.

The result of solving the subsystems in (7) is a set of joint queue length distributions at token arrival instants. To obtain the mean waiting time of a packet, we must obtain the mean queue lengths at the packet departure instants, or equivalently, at the token release instants. As shown by Kuehn [Kue79a], the queue lengths at the packet departure instants may be computed from the queue lengths at the token arrival instants. Kuehn derives an expression for the mean number of packets left by a departing packet using generating functions; here, we explain the expression using conditional probability.

Let p_{0i} denote the probability that a token finds no packets awaiting transmission on arrival at station i. Let $E[N_i]$ denote the mean queue length computed from the numerical solution of the steady state equations, and $E[N_i^*]$ denote the mean number of packets left behind by a departing packet. The mean number of packets left behind is simply the mean number found when the token arrives, given that there is at least one, together with the number that arrive at the station while service is being rendered, less the one served. From Kuehn [Kue79a],

$$E[N_i^*] = E[N_i]/(1 - p_{0i}) - 1 + \lambda_i E[S] \tag{9}$$

where λ_i is the packet arrival rate at station i and S is the time the token is held there. The mean response time of a packet (that is, its mean time between arrival and completion of its transmission) may be obtained from (9) by Little's law. Notice that p_{0i} may be computed directly from the equation for the mean cycle time c given by Kuehn,

$$c = kd / (1 - \sum_{j=1}^{k} \lambda_j E[S]) \tag{10}$$

since

$$p_{0i} = 1 - \lambda_i kd / (1 - \sum_{j=1}^{k} \lambda_j E[S]) \tag{11}$$

$$= 1 - c\lambda_i$$

This is useful for determining if the state space size G chosen above is large enough; if it is not, the values of the p_{0i}'s obtained by solving equation (7) will be too large. The imbedded Markov chain procedure is outlined in Figure 1.

Algorithm: Imbedded Markov Chain Analysis

1. Compute the state transition probability matrices $Q_0, Q_1, \ldots, Q_{k-1}$ using equation equations (5) and (6);
2. **For** each station i, $i = 0, 1, \ldots, k-1$ **do**
   Compute $Q_i Q_{i+1} \ldots Q_{k-1} Q_0 \cdots Q_{i-1}$;
   Using the power method, solve
   $q_i = q_i Q_i Q_{i+1} \cdots Q_{k-1} Q_0 \cdots Q_{i-1}$
   and compute marginal queue length distribution at token arrival instants for station i, yielding $E[N_i]$;
   Compute the mean queue length $E[N_i^*]$ from equation (9) and the packet response time form Little's Law.

Figure 1: Outline of Algorithm for Imbedded Markov Chain Analysis

The error of approximating an infinite state space with a finite one decreases as the maximum permitted number of queued packets at each station increases. Suitable state spaces for modeling token rings with moderate or heavy loads are extremely large; their solution are prohibitively costly in both space and time. If the system is symmetric, all the Q_i's (and consequently their cyclic products) are permutations of each other, so the storage and computational cost may be reduced considerably. Apart from the implicit assumption of lost packets, the probabilistic structure of the system is fully captured by the imbedding method. Some numerical results of the imbedded Markov chain analysis analysis will be presented in Section 5.

In the foregoing discussion, we have described the analytic difficulties caused by the periodicity of a token ring's imbedded Markov chain. We have shown how the distribution of packet queue lengths at token arrival instants may be used to obtain good approximations to the packet waiting times. Unfortunately, the analysis must be performed numerically. Moreover, it can only be used to study small systems under light and moderate loads because of its immense computational cost.

3. Kuehn's Method

Kuehn's method [Kue79a] attempts to avoid the computational cost of the direct imbedded Markov chain method described in the previous section by expressing mean packet queue lengths $E[N_i]$ at token arrival instants in terms of approximate formulae for the moments of the token's cycle time distribution. The expression for $E[N_i]$ may then be used in equation (9) to calculate the approximate mean queue length at token departure instants, and hence the packet's response time.

Under the assumption of independence of packet queue lengths, the Laplace transform (LT) of the cycle time distribution is given by

$$\varphi_C(s) = \prod_{i=0}^{k-1} \varphi_{U_i}(s) \prod_{i=0}^{k-1} (\alpha_i \varphi_{H_i}(s) + \left[1-\alpha_i\right]) \qquad (12)$$

where $\varphi_{U_i}(s)$ is the LT of the token passing overhead, $\varphi_{H_i}(s)$ is the packet transmission time, and α_i is the probability that at least one packet is awaiting transmission at the instant the token arrives at station i [Kue79a, Has72a].

The independence assumption that permits the multiplication of the Laplace transforms in equation (12) is clearly erroneous, and will result in the serious underestimation of the cycle time variance. Kuehn's approach is to reduce the error of (12) by is to condition the cycle time distribution on the event that a packet is absent or present at host j during a cycle, while assuming that these events are independent at all stations. Thus, equation (12) becomes

$$\varphi_C(s) = (1-\alpha_j)\varphi_{C'_j}(s)+\alpha_j\varphi_{C''_j} \tag{13}$$

where C'_j is the cycle time given that a job is present at station j when the token arrives there, and C''_j is the cycle time otherwise. The Laplace transforms of the C's may all be derived from the principles used to derive equation (12). The conditional cycle time variance may then be obtained from equation (13) by differentiation. The expression given by Kuehn for the mean number of queued packets at token arrival instants when all stations have the same arrival rate is

$$E[N_0] = p_0\lambda \frac{\lambda c'^{(2)}(1-\lambda c'')+c'(\lambda^2 c''^{(2)}+2-2\lambda c'')}{2(1-\lambda c'')^2} \tag{14}$$

where p_0 is given in equation (11) with $i=0$, c', c'' are the means of the conditional cycle times whose Laplace transforms were used in equation (13), and $c'^{(2)}$, $c''^{(2)}$ are the corresponding second moments.

The foregoing discussion shows how Kuehn has attempted to circumvent the intractability of an imbedded Markov chain analysis by constructing the refined approximation to the Laplace transform of the token cycle time distribution shown in equation (13). The independence assumption implicit in this equation will cause the underestimation of the cycle time variance and hence the mean number of queued packets, particularly when the packets have highly variable length. However, as will be shown in Section 5, the error caused by the independence assumption when packet lengths are fixed is very small. Thus, for token rings with low token passing overhead and fixed packet lengths, Kuhen's approximation method is both fast and reasonably accurate.

4. The Berry-Chandy Heuristic and a Proposed Modification

4.1. The Heuristic

The Berry-Chandy heuristic [Ber83a] is radically different from Kuehn's method, as it makes no use of the properties of imbedded Markov chains. Instead, it assumes that the total number of packets queued at all stations behaves as an M/G/1 queue, and has expected value

$$E[Q] = \rho^2(1 + C^2)/2(1 - \rho) \tag{15}$$

where C is the coefficient of variation of the packet length, and ρ is the sum of the ring utilizations over all stations,

$$\rho = \sum_{i=1}^{k} \lambda_i E[S]\,. \tag{16}$$

Here, λ_i denotes the arrival rate at station i. Overhead is ignored in these two equations. The packet waiting time at each station is found by apportioning the total mean queue length $E[Q]$ among all stations. The procedure for doing this is based on Berry and Chandy's *proportionate error hypothesis*, which states the error in an estimate of a performance metric is proportional to the metric's true value. The hypothesis was adapted from [Cha82a]. The outline of the Berry-Chandy algorithm given in Figure 2 is adapted from [Ber83a]; its derivation will not be repeated here.

Algorithm BCHAN:

Inputs:

First and second moments of the packet length distribution
Ring bandwidth
Number of stations k
Arrival rates λ_i, $i = 1,...,k$

Initialisation:

Set all mean waiting times and queue lengths to 0.0:
$E[Q_i] = 0.0$
$E[W_i] = \lambda_i E[Q_i]$
$LAST\_E[Q_i] = 0.0$
Since no customers have been apportioned to any station initially,
$w_i' = 0.0$
$q_i' = 0.0$
Compute the utilizations:
$\rho_i = \lambda_i E[S]$
$\rho = \Sigma_i \rho_i$
Compute CV of service:
$C = \sigma_S / E[S]$
where σ_S is the standard deviation of the packet length.

Step 1: Compute the overall mean queue length $E[Q]$ using the P-K formula (15) and the expression for the utilization (16).

Step 2: Compute the time taken for the token to reach station j given that it is held by station i for all i and j:

$$E[T_{ij}] = E[S^2]/2E[S] + d, \; i = pred(j)$$

$$= E[T_{ipred(j)}] + \rho_{pred(j)} E[S] + d, \; i \neq pred(j)$$

where $pred(j)$ is the index of the station preceding j in cyclic order.

Step 3: Compute the time taken for the token to reach j regardless of location:

$$E[K_j] = \Sigma_i \rho_i E[T_{ij}]$$

Step 4: Compute the work to be done at station j before the packet enters the ring:

$$E[Z_j] = E[S] * (E[Q_j] + \Sigma_{m \neq j} min(E[Q_j], E[Q_m] + \lambda_m E[W_j]))$$

Step 5: Compute the estimated wait times and queue lengths at all stations:

$$w_j' = E[K_j] + E[Z_j]$$
$$q_j' = \lambda_j w_j'$$

Step 6: Store the old values of $E[Q_i]$ in $LAST\_E[Q_i]$ for all i.
Apportion the queue lengths amongst the stations using

$$E[Q_i] = E[Q] q_i' / \Sigma_j q_j'$$

and compute the waiting times using

$$E[W_i] = \lambda_i E[Q_i]$$

Step 7: Compare $E[Q_i]$ and $LAST\_E[Q_i]$ for all i.
If the convergence criterion is met, terminate.
Otherwise, go to Step 4.

Figure 2: Berry-Chandy Token Ring Algorithm

A notable feature of the invariant (15) is that it does not include the token passing overhead in the utilization. Thus, the total number of packets present may be underestimated. From the apportionment step of the algorithm (Step 6), it is clear that the token passing overhead disappears from the analysis when arrival rates are the same at all stations, because the mean queue lengths will all be equal. The absence of overhead from the value of ρ also gives rise to a misleading stability condition $\rho<1$, where ρ is given in equation (16). According to Kuehn [Kue79a], the non-exhaustive service policy places a stability condition on each queue separately:

$$\lambda_j < \frac{1-\rho+\lambda_j E[S]}{kd+E[S]} \tag{17}$$

If this condition holds for all j, we obtain the stability condition

$$\sum_{j=1}^{k} \lambda_j (E[S]+d) < 1 \tag{18}$$

This condition states that the total packet throughput of the ring is restricted by the time taken for the token to pass from one station to the next, d as well as by the service time.

4.2. A Proposed Modification

The foregoing discussion shows that the Berry-Chandy heuristic implicitly overestimates the ring's capacity and hence that it will underestimate the total number of packets awaiting transmission. In this section, we propose modifications to correct these deficiencies.

One could interpret the utilization, or more specifically, the traffic intensity, as the proportion of time that the token is unavailable to serve a given waiting customer, rather than the proportion of time that work is being done. Each packet sent incurs a minimum of one unit of overhead d as implied by the stability condition (18). When the ring is nearly saturated, the token must spend approximately time d (the passing time) moving from one station to the next for each packet sent. As the load on the ring decreases, the number of times the token is seized in a given observation period and hence the token passing overhead delaying the transmission of each newly arrived packet increases. This overhead always lies between d and kd, where k is the number of stations. Therefore, an obvious modification to the Berry-Chandy heuristic is to replace equation (16) with

$$\rho = \sum_{i=1}^{k} \lambda_i (E[S] + d). \qquad (19)$$

This will make the Berry-Chandy heuristic's stability condition agree with Kuehn's.

Kuehn [Kue79a] has argued that the variance of the token cycle time also increases as the packet arrival rates decrease. Ideally, we would like to incorporate these load-dependent features of the overhead into the invariant.

As we saw in Section 3.3, accounting for the cycle time variance is not straight-forward because the probability of a packet's awaiting transmission at one station depends the number of stations with waiting packets during a cycle of the token. Accounting for the average overhead per packet is simpler. A packet arriving at an empty ring will have to wait for the token to complete between 1 and k empty scans before being dispatched. Overhead should be "charged" to the packet accordingly. We shall attempt to characterise this overhead in terms of the current ring loading. We conjecture that the overhead is a function that increases at the same rate as the probability of the token's performing an empty scan at a station. From (11), the probability of an empty scan at station i is given by

$$p_{0i} = 1 - \lambda_i kd / (1 - \sum_{j=1}^{k} \lambda_j E[S])$$

To simplify the discussion, assume that $\lambda_i = \lambda$, $i = 1,...,k$. Then we may write $p_{0i} = \beta$ say, for all i. It is easy to show that β is a decreasing convex function of λ. If the conjecture about the chargeable overhead is correct, the number of empty token scans seen by a newly arrived packet should also be a convex decreasing function of the total arrival rate, bounded above by kd and below by d. While we cannot determine what this function is, we know that it must have certain boundary values. At saturation, each packet will be charged exactly one unit of overhead. Near zero loading, the packet may be charged up to k units of overhead. A convex function that is decreasing on $[0,1]$ is $cos(\rho\pi/2)$, which vanishes when $\rho=1$. It follows that the following expression,

$$d[(k-1)cos((E[S] + d)\Lambda\pi/2) + 1]$$

with $\Lambda = \sum_{i=1}^{k}\lambda_i$, satisfies the stated convexity and boundary conditions: the first term is a convex functin of the loading, while the second term ensures that each packet is always charged for at least one empty scan. The effective traffic intensity may then be estimated by

$$\rho = \Lambda(E[S] + d[(k-1)cos((E[S] + d)\Lambda\pi/2) + 1]) \qquad (20)$$

Notice that this formula charges each packet with one unit of overhead at maximum load, and k units of overhead as zero load is approached. It is also in agreement with Kuehn's condition for the ring's stability.

The numerical results in Section 3.5 show that the original Berry-Chandy heuristic underestimates response times in all cases. This is not surprising, as its stability condition allowed more traffic than the correct one. The results also show that the simple modifications of the utilization ρ proposed above make the heuristic quite accurate over the range of parameter values tested, provided that both the packet length and token passing overhead per station are fixed.

5. Numerical Results

All three analyses discussed in the preceding sections have been tested with identical sets of arrival and service parameters. Following the notation of [Bux81b], the (fixed) packet transmission time was taken to be

$$T_p = \frac{(l_h + l_d)}{\nu} + k\,(d_a \tau + \frac{1}{\nu})$$

where l_h is the header length in bits, l_d is the data length, ν is the line speed in bits per second, k is the number of stations, and d_a is the distance between neighbouring stations (assumed constant). The token passing time between neighbouring stations was taken to be

$$T_t = \frac{l_t}{\nu} + \frac{1}{\nu} + d_a \tau$$

where l_t is the token length. These formulae assume a 1 bit delay at each station. For all runs,

$l_h = 112 \; bits$

$l_d = 1000 \; bits$

$l_t = 24 \; bits$

$\nu = 1 \; Mbits/sec$

$d_a = 40 \; metres$

$\tau = 5{\times}10^{-9} \; sec/metre$

For our analysis, the mean service time $E[S]$ was equated with T_p and the token passing overhead d was equated with T_t.

The results are compared in Tables 1-4. Figures obtained using the Berry-Chandy heuristic are displayed in the rows marked 'BC Orig.','BC Mod.1', and 'BC Mod.2'. Those marked 'BC Orig.' and 'BC Mod.1' were computed using $\rho = \Lambda E[S]$ and $\rho = \Lambda(d + E[S])$ respectively;. those marked 'BC Mod.2' were computed with ρ

given by equation (20). The rows labelled 'Imb.MC' and 'Kuehn' were computed using the imbedded Markov chain analyser and Kuehn's method respectively. As mentioned above, the imbedded Markov chain analysis is more accurate when the load is light than when it is heavy; queue lengths, response times, and waiting times will all be slightly underestimated. Kuehn's method yields predictions for these measures that are usually a little larger than those obtained from the imbedded Markov chain analysis. Since the original Berry-Chandy heuristic predicts waiting times that are less than those predicted by the imbedded Markov chain analysis, we can assume that it will always underestimate these measures. Kuehn's method is more accurate than the Berry-Chandy method, especially at light loads.

6. Conclusion

The immense storage requirements and computational cost of the approximate imbedded Markov chain analysis underscore the need for fast approximation schemes for modeling token rings. Accurate approximations are difficult to obtain because of the mutual dependence between queue lengths and because the token passing overhead results in a repeated portion of time during which packets may be awaiting dispatch but cannot be served.

The Berry-Chandy heuristic underestimates packet waiting times, particularly at light loads. A simple scheme for charging packets for the token passing overhead has made their approximation more accurate, at least for modeling rings with fixed packet length. Kuehn's method usually yields waiting time estimates that are slightly larger than those predicted by the imbedded Markov chain analysis, but will probably not be accurate when the packet length variance is large, because of the independence assumptions implied by its formulation.

Acknowledgments

The author wishes to acknowledge the valuable comments of Peter J. Denning and Herbert D. Schwetman. This work formed part of the author's doctoral dissertation [Bon84a].

References

Ber83a. Berry, R. and Chandy, K. M., "Performance Models of Token Ring Local Area Networks," *Proc. ACM Sigmetrics Conference,* p. 266-274 (1983).

Bon84a. Bondi, A. B., "Incorporating Open Queueing Models into Closed Queueing Network Algorithms," Ph.D. thesis, Dept. of Computer Sciences, Purdue University, West Lafayette, IN 47907 (May 1984).

Bur71a. Burge, W.H. and Konheim, A.G., "An accessing model," *J. ACM* **18** p. 400-404 (1971).

Bux81b. Bux, W., "Local area subnetworks: a performance comparison," *COM-29,* (10) p. 1645-1673 (1981).

Bux81a. Bux, W., Closs, F., Janson, P.A., Kuemmerle, K., and Mueller, H.R., "A reliable token-ring system for local-area communication," IBM Research Report no. RZ 1095 (#39517), IBM Zurich Laboratory, 8803 Rueschlikon/ZH, Switzerland (9/3/81).

Cha77a. Chandy, K. Mani, Howard, J.H., and Towsley, D.F., "Product Form and Local Balance in Queueing Networks," *Journal of the ACM* **24**(2) p. 250-263 (1977).

Cha82a. Chandy, K. Mani and Neuse, D., "Linearizer: A Heuristic Algorithm for Queueing Network Models of Computing Systems," *Comms. ACM* **25**(2) p. 126-133 (1982).

Cof69a. Coffman, E.G., "Analysis of a Drum Input/Output Queue under Scheduled Operation in a Paged Computer system," *J. ACM* **16**(1) p. 73-90 (1969).

Cof73a. Coffman, E.G. and Denning, P.J., *Operating Systems Theory,* Prentice Hall (1973).

Cox65a. Cox, D.R. and Miller, H.D., *The Theory of Stochastic Processes,* Wiley, New York (1965).

Dah74a. Dahlquist, G. and Bjorck, A., *Numerical Methods,* Prentice Hall, Englewood Cliffs (1974).

Has72a. Hashida, O. and Ohara, K., "Line accomadation capacity of a communication control unit," *Rev. Electr. Comm. Laboratories, Nippon T&T Public Corp.* **20** p. 231-239 (1972).

Ios80a. Iosifescu, M., *Finite Markov Chains and Their Applications,* John Wiley & Sons, Chichester (1980).

Ken53a. Kendall, D.G., "Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain," *Ann. Math. Stat.* **24** p. 338-354 (1953).

Kle75a. Kleinrock, Leonard, *Queueing Systems, Volume 1: Theory,* John Wiley, New York (1975).

Kon74a. Konheim, A.G. and Meister, B., "Waiting lines in a system with polling," *J. ACM* **21** p. 470-490 (1974).

Kue79a. Kuehn, P.J., "Multiqueue systems with non-exhaustive cyclic service," *Bell Sys. Tech. J.*, p. 671-698 (1979).

Mac57a. Mack, C., "The efficiency of N machines uni-directionally patrolled by one operative when walking time is constant and repair times are variable," *J. Roy. Stat. Soc. (B)* **19**(1) p. 173-178 (1957).

Mac57b. Mack, C., Murphy, T., and Webb, N.L., "The efficiency of N machines uni-directionally patrolled by one operative when walking time and repair times are constant," *J. Roy. Stat. Soc. (B)* **19**(1) p. 166-172 (1957).

Sal82a. Salwen, Howard, "On Ring Architected Local Networks," Proteon Report, Proteon Associates, Inc., Waltham, MA 02154 (April 1982).

Str83a. Strole, N. C., "A local communications network based on interconnected token-access rings: a tutorial," *IBM J. of Res. & Dev.* **27**(5) p. 481-496 (1983).

Table 1: Symmetric Loading of a Ring with Two Stations

Packet arrival rate/station: 4.3875e+01/sec. Transmission time/packet: 1.1144e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	9.7789e-02	2.6498e-03	6.0394e-05	1.1748e-03
BC Mod.1	1.0000e-01	2.7778e-03	6.3311e-05	1.1777e-03
BC Mod.2	1.0218e-01	2.9075e-03	6.6268e-05	1.1807e-03
Imb. MC	9.7789e-02	3.939e-03	9.000e-05	1.2044e-03
Kuehn	9.7789e-02	3.9448e-03	8.9911e-05	1.2043e-03

Packet arrival rate/station: 1.3163e+02/sec. Transmission time/packet: 1.1144e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	2.9337e-01	3.0448e-02	2.3133e-04	1.3457e-03
BC Mod.1	3.0000e-01	3.2143e-02	2.4420e-04	1.3586e-03
BC Mod.2	3.0591e-01	3.3706e-02	2.5608e-04	1.3705e-03
Imb. MC	2.9337e-01	3.520e-02	2.67e-04	1.382e-03
Kuehn	2.9337e-01	3.6171e-02	2.7480e-04	1.3892e-03

Packet arrival rate/station: 2.1938e+02/sec. Transmission time/packet: 1.1144e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	4.8894e-01	1.1695e-01	5.3309e-04	1.6475e-03
BC Mod.1	5.0000e-01	1.2500e-01	5.6980e-04	1.6842e-03
BC Mod.2	5.0782e-01	1.3099e-01	5.9709e-04	1.7115e-03
Imb. MC	4.8894e-01	1.321e-01	6.02e-04	1.7163-03
Kuehn	4.8894e-01	1.3329e-01	6.0760e-04	1.7220e-03

Packet arrival rate/station: 3.0713e+02/sec. Transmission time/packet: 1.1144e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	6.8452e-01	3.7131e-01	1.2090e-03	2.3234e-03
BC Mod.1	7.0000e-01	4.0833e-01	1.3295e-03	2.4439e-03
BC Mod.2	7.0703e-01	4.2656e-01	1.3889e-03	2.5033e-03
Imb. MC	6.8452e-01	4.210e-01	1.371e-03	2.455e-03
Kuehn	6.8452e-01	4.2510e-01	1.3841e-03	2.4985e-03

Table 1 continued.

Packet arrival rate/station: 3.9488e+02/sec. Transmission time/packet: 1.1144e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	8.8010e-01	1.8150e+00	4.0899e-03	5.2043e-03
BC Mod.1	9.0000e-01	2.0250e+00	5.1281e-03	6.2425e-03
BC Mod.2	9.0311e-01	2.1045e+00	5.3296e-03	6.4440e-03
Imb. MC*	8.8010e-01	n.a.	n.a.	n.a.
Kuehn	8.8010e-01	2.0797e+00	5.2667e-03	6.3811e-03

*Accurate results unavailable because of storage limitations

Table 2: Symmetric Loading of a Ring with Three Stations

Packet arrival rate/station: 2.9219e+01/sec. Transmission time/packet: 1.1156e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	9.7790e-02	1.7666e-03	6.0460e-05	1.1761e-03
BC Mod.1	9.9999e-02	1.8518e-03	6.3377e-05	1.1790e-03
BC Mod.2	1.0436e-01	2.0268e-03	6.9365e-05	1.1850e-03
Imb. MC	9.7790e-02	2.956e-03	1.01e-04	1.217e-03
Kuehn	9.7790e-02	3.0011e-03	1.0271e-04	1.2183e-03

Packet arrival rate/station: 8.7658e+01/sec. Transmission time/packet: 1.1156e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	2.9337e-01	2.0300e-02	2.3158e-04	1.3472e-03
BC Mod.1	3.0000e-01	2.1429e-02	2.4446e-04	1.3601e-03
BC Mod.2	3.1181e-01	2.3546e-02	2.6861e-04	1.3842e-03
Imb. MC	2.9337e-01	2.565e-02	2.93e-04	1.408e-03
Kuehn	2.9337e-01	2.4526e-02	2.7980e-04	1.3954e-03

Packet arrival rate/station: 1.4610e+02/sec. Transmission time/packet: 1.1156e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	4.8895e-01	7.7970e-02	5.3369e-04	1.6493e-03
BC Mod.1	5.0000e-01	8.3333e-02	5.7040e-04	1.6860e-03
BC Mod.2	5.1562e-01	9.1478e-02	6.2615e-04	1.7418e-03
Imb. MC	4.8895e-01	9.118e-02	6.24e-04	1.740e-03
Kuehn	4.8895e-01	8.6113e-02	5.8943e-04	1.7050e-03

Packet arrival rate/station: 2.0454e+02/sec. Transmission time/packet: 1.1156e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	6.8454e-01	2.4757e-01	1.2104e-03	2.3260e-03
BC Mod.1	7.0000e-01	2.7222e-01	1.3309e-03	2.4465e-03
BC Mod.2	7.1404e-01	2.9716e-01	1.4529e-03	2.5685e-03
Imb. MC	6.8454e-01	2.794e-01	1.366e-03	2.482e-03
Kuehn	6.8454e-01	2.7013e-01	1.3207e-03	2.4363e-03

Table 2 continued.

Packet arrival rate/station: 2.6297e+02/sec. Transmission time/packet: 1.1156e-03 sec. Token passing overhead per station: 2.5200e-05 sec.				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	8.8012e-01	1.0769e+00	4.0951e-03	5.2107e-03
BC Mod.1	9.0000e-01	1.3500e+00	5.1335e-03	6.2491e-03
BC Mod.2	9.0622e-01	1.4595e+00	5.5499e-03	6.6655e-03
Imb. MC*	8.8012e-01	n.a.	n.a.	n.a.
Kuehn	8.8012e-01	1.3585e+00	5.1658e-03	6.2814e-03

*Accurate results unavailable because of storage limitations

Table 3: Ring with Asymmetric Loading 0.5

Input Parameters	
Transmission time/packet:	1.1156e-03 sec.
Token passing overhead per station:	2.5200e-05 sec.
Arrival rate at station 0:	4.3829e+01 pk/sec
Arrival rate at station 1:	8.7658e+01 pk/sec
Arrival rate at station 2:	3.0680e+02 pk/sec

Performance measures for station 0				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	4.8896e-02	1.4773e-02	3.3705e-04	1.4527e-03
BC Mod.1	5.0000e-02	1.5571e-02	3.5527e-04	1.4709e-03
BC Mod.2	5.1562e-02	1.6750e-02	3.8216e-04	1.4978e-03
Imb. MC	4.8896e-02	1.833e-2	4.4e-04	1.56e-03
Kuehn	4.8896e-02	1.8635e-02	4.2518e-04	1.5408e-03

Performance measures for station 1				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	9.7791e-02	3.4934e-02	3.9853e-04	1.5141e-03
BC Mod.1	1.0000e-01	3.6963e-02	4.2168e-04	1.5373e-03
BC Mod.2	1.0312e-01	3.9983e-02	4.5612e-04	1.5717e-03
Imb. MC	9.7791e-02	4.0810e-02	4.9e-04	1.61e-03
Kuehn	9.7791e-02	4.3508e-02	4.9632e-04	1.6119e-03

Performance measures for station 2				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	3.4227e-01	1.8420e-01	6.0040e-04	1.7160e-03
BC Mod.1	3.5000e-01	1.9747e-01	6.4363e-04	1.7592e-03
BC Mod.2	3.6093e-01	2.1770e-01	7.0959e-04	1.8252e-03
Imb. MC	3.4227e-01	2.0863e-01	7.1e-04	1.82e-03
Kuehn	3.4227e-01	2.1584e-01	7.0352e-04	1.8191e-03

Table 4: Ring with Asymmetric Loading 0.6

Input Parameters	
Transmission time/packet:	1.1156e-03 sec.
Token passing overhead per station:	2.5200e-05 sec.
Arrival rate at station 0:	4.3829e+01 pk/sec
Arrival rate at station 1:	1.7532e+02 pk/sec
Arrival rate at station 2:	3.0680e+02 pk/sec

Performance measures for station 0				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	4.8896e-02	1.8394e-02	4.1968e-04	1.5353e-03
BC Mod.1	5.0000e-02	1.9431e-02	4.4334e-04	1.5589e-03
BC Mod.2	5.1298e-02	2.0705e-02	4.7241e-04	1.5880e-03
Imb. MC	4.8896e-02	2.485e-02	5.6698e-04	1.6826e-03
Kuehn	4.8896e-02	2.3928e-02	5.4593e-04	1.6615e-03

Performance measures for station 1				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	1.9558e-01	1.0804e-01	6.1625e-04	1.7319e-03
BC Mod.1	2.0000e-01	1.1544e-01	6.5848e-04	1.7741e-03
BC Mod.2	2.0519e-01	1.2473e-01	7.1145e-04	1.8271e-03
Imb. MC	1.9558e-01	1.3842e-01	7.8953e-04	1.9051e-03
Kuehn	1.9558e-01	1.4664e-01	8.3646e-04	1.9521e-03

Performance measures for station 2				
Method	Eff. Util.	Que.Len. Waiting	Waiting Time	Response Time
Orig. BC	3.4227e-01	2.9010e-01	9.4557e-04	2.0612e-03
BC Mod.1	3.5000e-01	3.1513e-01	1.0271e-03	2.1427e-03
BC Mod.2	3.5909e-01	3.4744e-01	1.1325e-03	2.2481e-03
Imb. MC	3.4227e-01	3.1540e-01	1.0280e-03	2.1436e-03
Kuehn	3.4227e-01	3.1316e-01	1.0207e-03	2.1363e-03