Extensibility and Limitations of FDDI

David Game
Kurt Maly
Department of Computer Science
Old Dominion University
Norfolk, Virginia 23529-0162

March 6, 1990

Abstract

Recently two standards for MANs, FDDI and DQDB, have emerged as the primary competitors for the MAN arena. Great interest exists in building higher speed networks which support large numbers of node and greater distance, and it is not clear what types of protocols are needed for this type of environment. There is some question as to whether or not these MAN standards can be extended to such environments.

This paper investigates the extensibility of FDDI to the the Gbps range and a long distance environment. It does this first by showing which specification parameters affect performance and providing a measure for predicting utilization of FDDI. A comparison of FDDI at 100Mbps and 1Gbps is presented. Some specific problems with FDDI are addressed and modifications which improve the viability of FDDI in such high speed networks are investigated. [1]

[1]This work was supported by CIT grant RF-89-002-01, NASA grant NAG-1-908 and Sun Microsystem grant RF 596043.

1 Introduction

Network data rates are commercially available at rates in the 100Mbps per channel class. The two most prominent of those are competing MAN standards FDDI[20] and DQDB(QPSX)[5] and much research is currently ongoing in an effort to better understand their performance capabilities and limitations[8,2]. Research, however, is going forward and a national research initiative is underway to develop Gbps networks to be employed as a backbone for a national research network[9].

Many questions still remain as to the approach which can best suit the requirements of such a network. A national network will likely transport synchronous and asychronous traffic, support large numbers of nodes (at least 100 and likely over 1000), and be spread over very large distances (over 1000 Kilometers). The impact of considering these types of parameter ranges can be very negative for token rings due to increased token cycle time and for CSMA/CD due to the increased slot times. No known research exists to show how FDDI and DQDB are affected. Current national networks are very slow packet networks on the order of 56Kbps. As we move towards Gbps speeds, the networks will be much more expensive and efficiency will become a much more important factor.

Increased data rates could be accomplished by focusing on the development of transmitter/receiver devices which are capable of functioning at such high rates[7, 11,16,18], i.e. just build gigbit speed lasers. There are of course numerous problems associated with such high speed devices other than the transmitter/receivers themselves, such as how to build computers which can process data at the rate of the network and what types of protocols would work best at these rates. For example, [9] suggests that it might be necessary to structure packet sizes to be large in order to minimize overhead impact.

Another approach is to examine how current transmitter/receiver technology can be optimized to improve characteristics such as throughput and delay. Parallel channels show some promise for improved performance [13,12,14] and is the subject of some research. Many architectures have been proposed in an attempt to design more efficient networks. Strategies have included 'train' protocols [22,21,10], hybrid CSMA/CD protocols[3,15,13,12,4], slotted and register insertion rings[6], and numerous others.

In this paper I will examine the viability for scaling FDDI, a 100Mbps token ring protocol, to the type of environment mentioned above. A number of problems will be defined and a suggestion for performance improvement will be given. The suggestion for performance enhancement is applicable to token ring networks at any speed and distance and provides a framework for improving other types of networks.

2 FDDI

2.1 Basic Token Ring

A token ring network is distinguished by the manner in which transmission rights are granted to nodes on the network. The token packet circulates on the ring, passing by each node. If a node's queue is empty, it simply continues circulating the token to the next node. However, if the node has a message in its queue to transmit, it will effectively remove the token by changing some information in the packet. The token itself is not actually removed, but is transmitted in some altered form such that subsequent nodes will not see it as a token and will not attempt to transmit. The modified token continues circulating and is eventually consumed (not retransmitted) when it reaches the node which is in the process of transmitting (holding the token).

Figure 1 illustrates how the token is 'removed'. Node 4 has a message for node 2 and is waiting for the token before initiating transmission. Figure 1.*b* shows that the token has been modified so that it will not be recognized as such and the message has been placed on the network. Part *c* shows that the token has been retransmitted on the network and is available for subsequent use.

As the token continues around the ring, subsequent nodes remove the token and append another packet. This scenario is illustrated in Figure 2 where node 1 has a message for node 5. In order to send the message, the token is modified, the message for node 5 is transmitted and a new token is made available to the other nodes. The network becomes filled with messages and *old tokens*. At most one real token is on the network at any instant of time.

The exact time at which the *old tokens* and messages are removed is dependent upon the distance of the network, the length of the packet and the data rate of the network. The most important factor to note is that all nodes except the node holding the token are forwarding messages. The node holding the token does not forward the incoming message but instead forwards its own message. Incoming messages are lost. Eventually, the *old token* will encounter a node which is in the process of transmitting a message and be removed.

Data packets are removed in a slightly different manner. It is important that they be removed by the sender so that a receiver will not accept the message a second time if it recirculates. The tail of the message is removed(modified) at the sender once the address is recognized, whereas the fragments of the headers of these packets are removed as mentioned above[19].

2.2 Token Rotation Time

The decentralized access mechanism of a token ring protocol can place limitations on how long a station has access to the network once it obtains the token. The

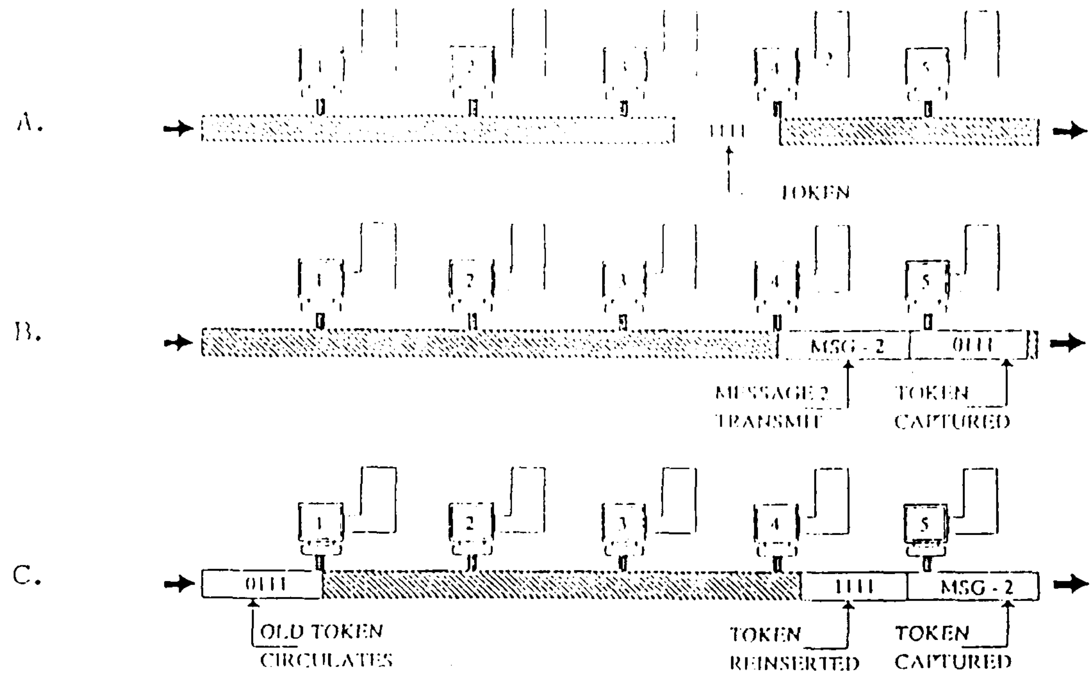

Figure 1: Token Ring Protocol Token Capture

approach in FDDI is *to* limit the amount of *time* for which one staion can hold the token[23]. Each node has a timer which is reset when the token arrives. When the token returns, the node may capture it only for an amount of time which will assure that the token will return within a specific time period. This time period is the *Target Token Rotation Time*, *TTRT*. The value of this parameter is negotiated amongst all nodes on the network and is essentially the smallest value selected by any node. This defines the maximum amount of time between access to the network and should provide for synchronous traffic.

Although it can be shown that the algorithm will guarantee that the token will return within the negotiated time frame on the average[8], it can not be guaranteed that the node will be able to hold the token at all once it returns. This has serious implications for periodic traffic and maximum throughput and will be examined in the next section.

3 Impact of Token Rotation Time

Johnson[8] provides analysis concerning the timing requirements of FDDI for both the ideal and the non-ideal(including overhead) cases. For the ideal case, the token can always be guaranteed to return to the station within $2 * TOPR$ where $TOPR$ is the current operating value of the $TTRT$, within which the token should return. For example, if the currently negotiated value of TOPR is 125 μseconds, then the token can only be guaranteed to return within 250 μseconds. It would appear then that one would simply negotiate for one-half of the desired $TTRT$ and then the proper availability of the token could be assured. For reasons of maximizing utilization of

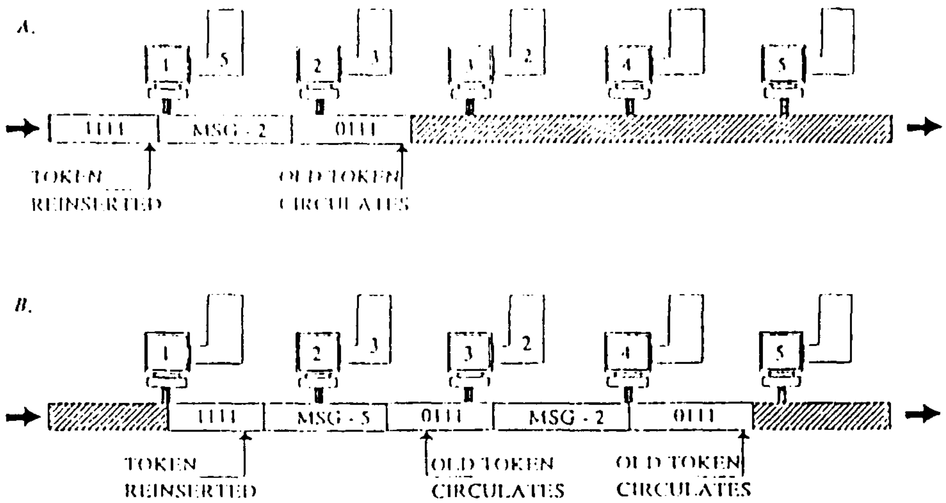

Figure 2: Token Ring Protocol Without Removal at Destination

the network, there is compelling motivation to have a large value of $TTRT$ thus resulting in a tradeoff between the design objective to support synchronous traffic and the need for high utilization. The following section illustrates the impact of $TTRT$ on utilization.

3.1 Parameters affecting TTRT

As cited in [8], the primary components of ring overhead are as follows.

- Total Propagation Delay (D_{prop}) is determined by multiplying the propagation delay for fiber optic media (5085 ns/km) by the length of the network.
- Latency (L_n) occurs at each node and is effectively the delay between the time a bit arrives as a node and departs the node. Therefore, if one examines the round trip of a single bit around the network, the delay is increased by the latency at each node times the number of nodes. L_{tot} represents the total latency of the network($L_{tot} = N \times L_n$).
- The number of nodes to capture the token, N_c, increases the delay of the token rotation. In the minimal delay case, no node needs the token and this component is nothing, but no information is transmitted. By focusing upon the process of transmitting a frame at a single node, this overhead becomes apparent. The head of the token arrives at the node and is passed on to the rest of the network while the node waiting to transmit identifies this as the token. Recognition takes place before the tail of the token is retransmitted, providing the capturing node the opportunity to modify the end of the token,

transforming it into a non-token frame, and thereby capturing the token. The delay required to accomplish this is incorporated into Latency, L_n. The node then proceeds to transmit its packet and *retransmit* the token to its neighboring node as explained in the section Basic Token Ring and Figure 2.

- As each node captures the token and retransmits it, an additional delay equal to the Token Transmission Time (T_t) will be incurred. Note that the delay from message transmission does not contribute to *overhead* delay.

- The design specifications of FDDI[20] allow for a maximum Transmitter Idle Time(T_i). This represents the time which is required between recognition of the token by the node and beginning of transmission of the frame. As in the previous item, this is only a factor when nodes are actually capturing the token.

Specified times for these delay components can be found in [20].

Latency per connection	600 ns
Token Transmission Time	880 ns
Max Transmitter Idle Time	3500 ns

Consider three scenarios for FDDI as a basis for evaluating the impact of these parameters:

1. 10 nodes separated by a distance of 100 meters each (1 km total) - representing a backbone for interconnecting local area networks,

2. 3 nodes separated by a distance of 10 meters each (30 meters total) - representing the connection of two mainframe/supercomputers or peripheral equipment which is in a close physical proximity,

3. 500 nodes each separated by a distance of 100 meters each (50 Km total) - representing a HSLDN or MAN.

Table 1 illustrates the delays[2] inherent in each of the scenarios. The difference between MAX and MIN TOTAL DELAY is the number of nodes transmitting on a token rotation.

One can see that as latency improves below 60ns, the effect will be significant only in the MIN TOTAL DELAY for mainframe environments.

[2] a one bit delay is equivalent to 10 ns for a 100Mbps network

Latency : 600 ns per node

	1.Backbone	2.Mainframes	3.HSLDN
Prop Delay (D_{prop})	$5.085\mu s$	$0.1526\mu s$	$2543\mu s$
Latency (L_{tot})	$6\mu s$	$1.8\mu s$	$300\mu s$
Max Token Trans (T_t)	$8.8\mu s$	$2.64\mu s$	$440\mu s$
Max Trans Idle (T_i)	$35\mu s$	$10.5\mu s$	$1750\mu s$
MAX TOTAL DELAY	$54.885\mu s$	$15.0926\mu s$	$5033\mu s$
MIN TOTAL DELAY	$11.085\mu s$	$1.9526\mu s$	$2843\mu s$

Latency : 60 ns per node

	1.Backbone	2.Mainframes	3.HSLDN
Prop Delay (D_{prop})	$5.085\mu s$	$0.1526\mu s$	$2543\mu s$
Latency (L_{tot})	$.6\mu s$	$.18\mu s$	$30\mu s$
Max Token Trans (T_t)	$8.8\mu s$	$2.64\mu s$	$440\mu s$
Max Trans Idle (T_i)	$35\mu s$	$10.5\mu s$	$1750\mu s$
MAX TOTAL DELAY	$49.485\mu s$	$15.0926\mu s$	$4763\mu s$
MIN TOTAL DELAY	$5.685\mu s$	$.3326\mu s$	$2573\mu s$

Table 1: Overhead Delay Parameters for FDDI Scenarios

3.2 TTRT vs Utilization

The purpose of this analysis is to develop a tool which is reasonably simple to use and which will be able to predict maximum utilization of an FDDI network. Practically all aspects of the derivation use average values of the random variables with focus given to heavily loaded conditions. Only asynchronous traffic is considered with justification for ignoring synchronous traffic in the analysis being given in the following section. The end of the analysis will provide results from a simulation of FDDI to illustrate the degree of accuracy of these approximations.

All nodes on a FDDI network use the same value of $TTRT$. If a node does not obtain the token in time to transmit its data and maintain the required timing restraints, it simply forwards the token to the next node. One way of viewing utilization (U) is to represented it as

$$U = \frac{TTRT - TRO}{TTRT} \tag{1}$$

where TRO represents the Token Rotation Overhead. During a single round trip of the token, only a certain percentage of the time can be spent sending data. The rest of the time essentially represents the amount of time required to transmit the token around the ring. Part of TRO is fixed and independent of the network load. All other factors remaining fixed, one can see that for a heavily loaded network, increasing $TTRT$ will increase utilization $\cdots$ at the expense of delay. Here we examine an estimation for utilization as a function of $TTRT$.

Using the terms developed in the previous sections, TRO can be expressed as follows

$$TRO = N_c \times (T_t + T_i) + D_{prop} + L_{tot} \tag{2}$$

The number of nodes to capture the token on a rotation is dependent upon the available bandwidth for transmission ($TTRT - TRO$), the packet length, load and the number of packets transmitted by each node which captures the token. All of the variables except N_c are static values, however, for this derivation, separate the components of TRO into two terms as follows, TRO_s representing the component of TRO which is independent of the number of packets transmitted and the dynamic part TRO_d.

$$TRO = TRO_s + TRO_d \tag{3}$$

$$TRO_s = D_{prop} + L_{tot} \tag{4}$$

$$TRO_d = N_c \times (T_t + T_i) \tag{5}$$

The dynamic component is determined by the number of nodes which capture the token. Each time the token is captured, there is a transmitter idle delay and a retransmission of the token. It is possible that a node can transmit multiple packets for a single token capture. If each node capturing the token transmits twice as many messages, the token retransmission(T_t) and transmitter idle delays(T_i) would occur half as often.

Consider the range of values which N_c has with the load uniformly distributed among the nodes. Under low loads, few nodes have messages to transmit. As network load increases, N_c increases, approaching the number of nodes on the network N, then it decreases as the network becomes overloaded. In the last case, as nodes have large queues, one token capture results in a large number of packet transmissions. Eventually, each node holds the token for a period of time which precludes other nodes on the network from capturing the token until its next rotation.

As the queues at each node overload, the utilization actually increases as there are fewer token captures per rotation; however, we are interested in determining the maximum traffic which the network can support without queue buildup. In such an overloaded situation, the network cannot support the traffic levels even though overall utilization may be higher. Therefore, it is assumed that traffic is distributed such that on the average a node only has a single packet(or less) to transmit per token rotation and that the maximum value of N_c is N. The number of packets which can be transmitted is dependent upon the number of packets which can be transmitted during $TTRT - TRO_s$. N_c would be defined as

$$N_{pt} = N_c = \frac{TTRT - TRO_s}{\frac{P}{R} + T_t + T_i} \tag{6}$$

where
P is the packet length
R is the transmission rate of the network and
and N_{pt} is the number of packets transmitted.

As the load increases, the number of nodes capturing the token would have a limit of

$$N_c = N \tag{7}$$

and the number of packets transmitted with maximum token captures N_{ptM} would be

$$N_{ptM} = \frac{TTRT - TRO_s - N \times (T_t + T_i)}{\frac{P}{R}} \tag{8}$$

Substituting Equation 8 back in to Equation 1 , U can be expressed as

$$U = \frac{TTRT - TRO_s - N \times (T_t + T_i)}{TTRT} \tag{9}$$

Figures 3 and 4 illustrate the predicted and real maximum utilization for the backbone and MAN scenarios listed above.

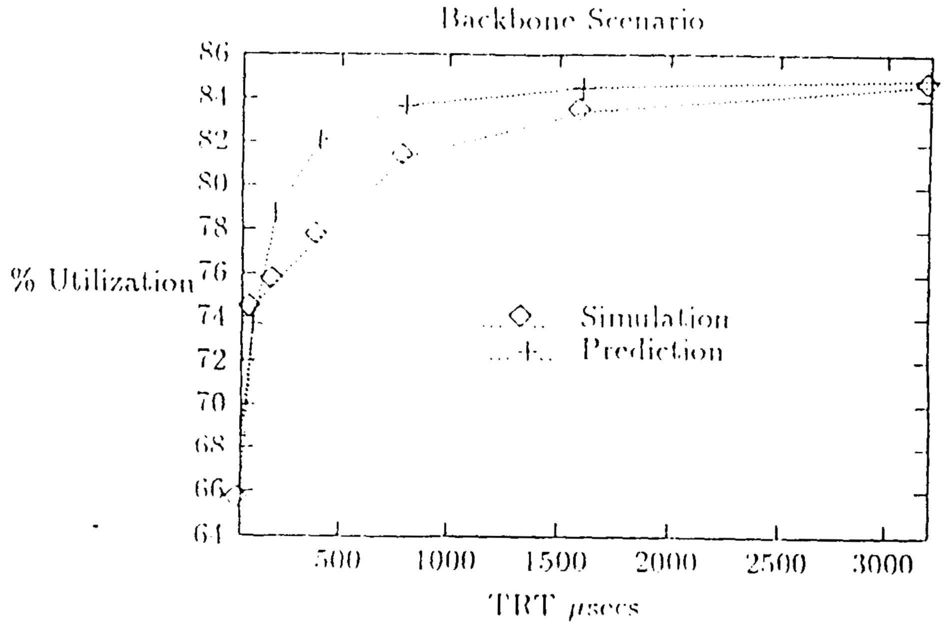

Figure 3: Backbone Predicted Utilization vs TRT

3.3 TTRT Impact on Synchronous Traffic

Synchronous traffic has not been included in the previous analysis. Because synchronous traffic by its nature places a uniform load on the system per token rotation, we can initially consider it as an overhead of the token rotation. After calculating the maximum utilization as described above, add the percentage of synchronous traffic to the previous utilization value to obtain the true utilization. The only approximation involved is due to the number of token captures which could be higher if, for example, synchronous traffic packets were small and distributed to a large number of nodes. This would further reduce maximum utilization.

Application of the previous results to typical synchronous traffic requirements indicates that FDDI does not support synchronous traffic without significant decrease in utilization. ISDN compatability requires that synchronous traffic must be delivered at the rate of once every $125\mu s$. In order to guarantee this arrival rate, a $TTRT$ of $62.5\mu s$ must be established. For a conservative estimate, assume that $TTRT$ is $125\mu s$, knowing that on the average $125\mu s$ will be attainable although in some instances packets may be lost due to the inability to guarantee $TTRT$ can be met. Comparing this with the lower node latency and ignoring packet overhead, the maximum utilization is illustrated in Table 2.

The table indicates that FDDI could not support an acceptable ISDN interface in most configurations and that it would be extremely unlikely that synchronous traffic with comparable periodicity could be supported in a long distance (MAN) environment. It is also interesting to note that in the scenario for a backbone, the

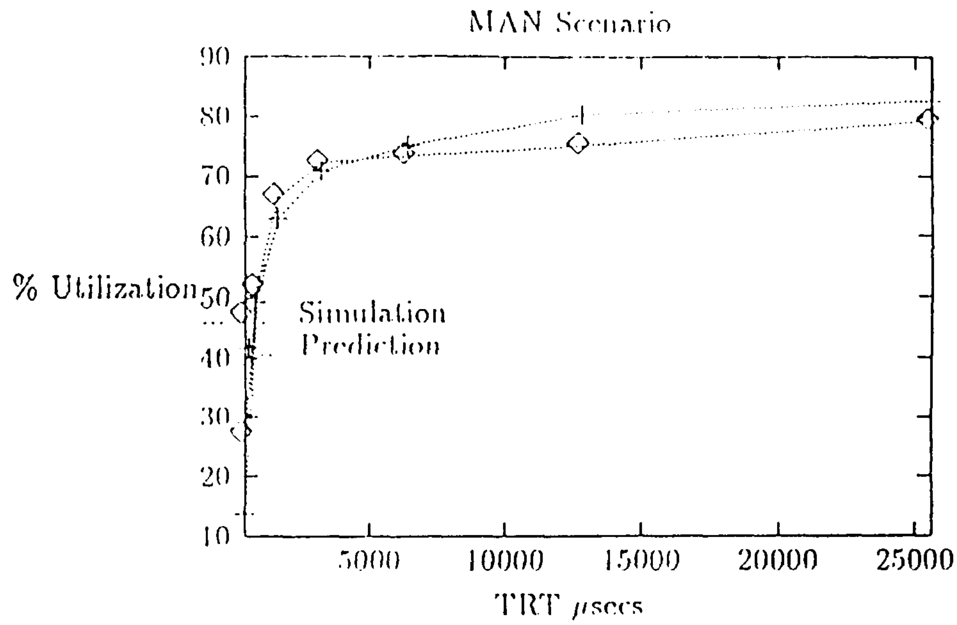

Figure 4: MAN Predicted Utilization vs TRT

	1.Backbone	2.Mainframes	3.HSLDN
MAX TOTAL DELAY	60%	88%	****
MIN TOTAL DELAY	95.5%	99.9%	****

Table 2: Maximum Utilization for 125μs $TTRT$

utilization could drop significantly depending upon the number of nodes which can capture the token on one rotation. This is of course also dependent on the packet size being transmitted.

4 Scaling FDDI to Gigabit Speeds

With the disparity between the development of the speed of transmission technology (optical systems) and the speed of processing elements in a computer, one might reasonably question the degree to which gigabit networks can be used at all. This disparity between the speed of transmitter technology and processing elements is tolerable if large numbers of nodes can be viewed as using the gigabit speeds if only for short durations. This is one reason to examine the scalability of FDDI in terms of the *number of nodes* it will support. The previous section also provided some insight as to the impact which number of nodes will have on performance. The second most important factor to examine is *network length*. Token ring networks are usually dependent upon short propagation delays for providing fast network access.

Given that this is not a detailed investigation, the two factors will not be addressed independently. Instead, the network configurations examined will look at increasing the number of nodes while maintaining the same internode distance, thereby increasing the network length simultaneously.

When considering the scalability of FDDI, one could view a scaling of the transmitter with proportional scaling of the speed of the nodes, or the transmitter speed could be scaled leaving the node processing speeds (values of Transmitter Idle Time, etc. discussed in the previous section) at the same rate. The data which follows is an examination of the latter given its greater probability of occurrence.

The parameter space examined here varies the number of nodes between 100 and 500 maintaining internode distance at 100m, resulting in network lengths between 10Km and 50Km. The $TTRT$ has been chosen at a level which allows for maximum utilization in the 75-90% range in general(10000 μs). The delay terms mentioned in the previous section were constant in all runs. Packet size was fixed at 1000 bits.

Figure 5 shows access delay for standard(100Mbps) and gigabit FDDI. The vertical axis scaling has been chosen to be twice $TTRT$(10000 μs). This shows how performance suffers dramatically in long-distance networks when more than one token rotation is required to deliver data. As expected, performance degrades in each graph as nodes and distance are increased. The two graphs reveal that gigabit FDDI begins to degrade at approximately 70%, whereas standard FDDI degrades slightly after 70%. One might expect that a faster transmission rate would mean shorter delays, however, these graphs tend to show comparable results. The reason is that with a constant packet size there are more packets at 70% load in gigabit FDDI, and more nodes are also transmitting. The previous section indicated the overhead associated with token capture and how it relates to the number of nodes.

The final graph in figure 5 combines the 100 and 500 nodes curves from the previous two graphs. Surprisingly, it does not appear that the scaling of the transmitter is the issue. Both speeds indicate similar performance curves. Number of nodes and distance are the factors which distinguish the shapes of the curves.

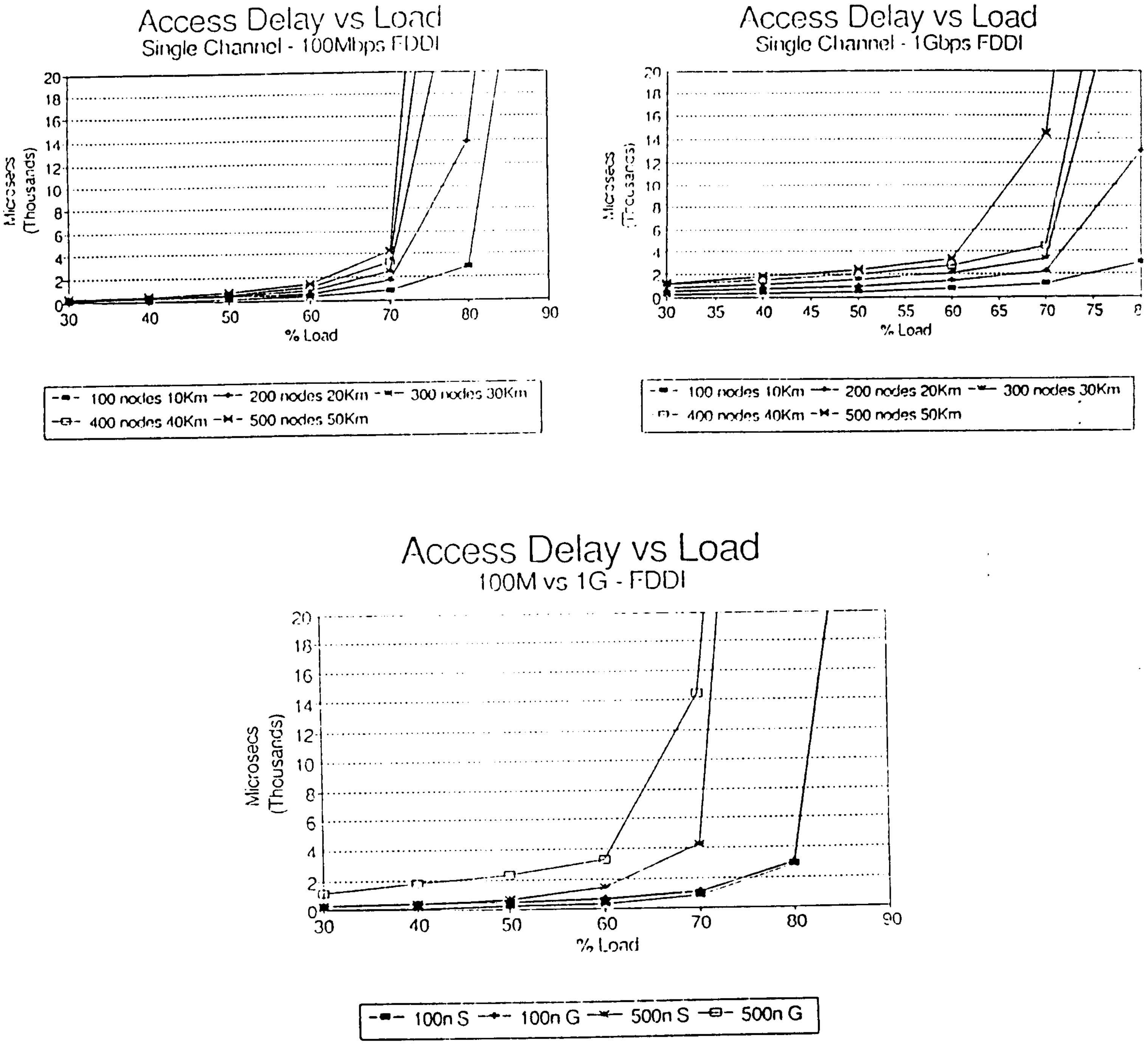

Figure 5: Standard FDDI vs 1Gbps FDDI

5 Performance Improvement

As the data transmission rates and distance covered by the networks increase, the number of simultaneous packets on a network increases dramatically. The parameter a where

$$a = \frac{length\ of\ data\ path}{length\ of\ packet} \tag{10}$$

represents this concept and frequently arises as a crucial parameter in network performance. For example, a network of 10 Mbps capacity, 2000 packet length and LAN length of 1 km can only hold .003125 packets at a time. Network designs with capacities such as 1 Gbps, 100 km and packet lengths of 2000 bytes, could contain approximately 31 packets simultaneously.

Increasing the number of packets on a network as shown above demands that greater attention be given to the management of these packets. Most LANs do not have this problem of simultaneous packets. Networks such as DQDB[17] which employs a slotted scheme or DSMA/RN[4] which uses a hybrid CSMA/CD technique may provide greater opportunity for optimization of this packet capacity.

In addition, the metrics for network performance, or at least our view of them, needs to be reconsidered. When analyzing the performance of a data communications network, one typically uses utilization as a metric of evaluation. If a 100Mbps network is capable of delivering 100Mbps of data, then the network is assumed to have 100% utilization (ignoring packet overhead).

This 100% limitation is based upon the assumption that only one node is transmitting at a time. If it is only possible for one node to transmit at a time as in most CSMA/CD networks or token ring networks, then this is a reasonable assumption. Even in situations where link level protocols provide for the existence of multiple packets on the network simultaneously as in *selective retransmission* and *go-back-n*, and in systems which allow for the building of a train of packets such as Expressnet, no consideration is given to the possibility of simultaneous transmission by nodes. Register insertion rings[6], DSMA/RN[4] and others[1] have shown that utilization greater than 100%(throughput greater than 1.0) is achievable.

Assume that nodes are numbered from 1,...,N and placed in a ring. If node 1 can send to node 2 while node 3 sends to node 4, 200Mbps is being transmitted. To understand the inefficiency of a ring running at close to 100% utilization, consider not only whether the network is filled with packets, but whether or not the packets are doing useful work, where work which is not useful occurs when packets take up network capacity but have already been delivered to the receiver.

The focus of the suggestion for improvement in performance of FDDI in this paper is on recovering the unused packet capacity by removing the packet at the destination and inserting new packets in their place which I will call *destination insertion. Destination insertion* will also allow for multiple simultaneous transmitters on the network and increased throughput. The technique is also applicable to

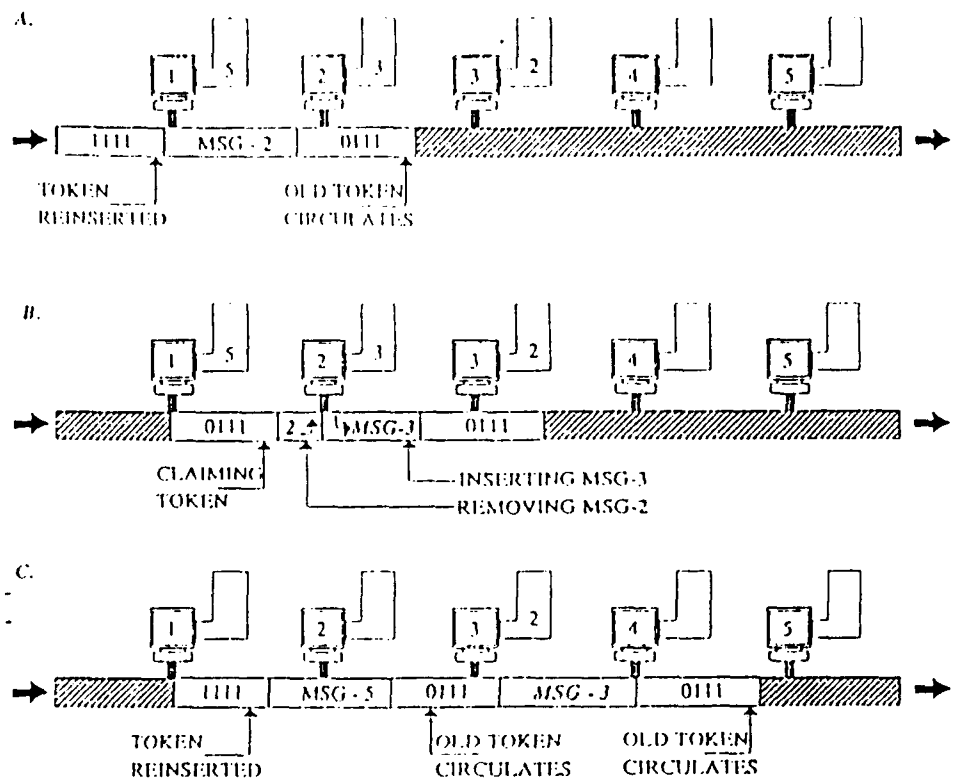

Figure 6: Removing Packet at Destination

DQDB and token rings in general, although the analysis does not address DQDB.

5.1 Destination Insertion

Figure 2 illustrates the removal of the message at the receiver. In the second part of this diagram, *MSG - 2* has reached the destination node and is about to arrive back to the original sender, node 4. It has been shaded to emphasize that the slot is only delivering an acknowledgment and that the capacity of the network is not being used in an optimal fashion. In this case the slot could have been used by node 2 to transmit the message to node 3. I.e., two messages could have been delivered instead on one with this packet-slot.

Figure 6 depicts node 2 removing the message from node 4 and at the same time inserting its message to node 3. The removal of the token at node 1 and transmission of the message from node 1 to node 5 is unaffected by this squeezing of the message from node 2 to node 3 into the train of packets. The squeezed data can not be longer than the message which it is replacing.

Destination insertion works as follows. When a node transmits a message on the network, the message proceeds until it reaches its destination. At that point it is marked as received and the slot containing the message is available for further use. As long as the slot is on the ring, it can be used by other nodes. This raises two additional questions:

- how long will the slot remain on the ring, and
- which nodes(messages) are candidates for reusing the slot.

5.2 Restrictions on slot reuse

This message would normally travel at least as far as the original sender and could conceivably travel even further if the original sender releases the token before the packet returns. If the time of transmission of the message is at least as long as the propagation delay of the network($a < 1$), the message will terminate at the sender. If the time of transmission is much less that the propagation delay($a > 1$), the token will leave the node before the message has made a loop and the message may not reach the node holding the token for a significant period of time.

When $a < 1$, the packet arrives at the sender before transmission is complete. Assume that node 1 holds the token and has a message for a node i. When node i receives the slot, it is reusable by any node up to node 1, but not beyond node 1. If an attempt is made to send a message past node 1, it will be absorbed by node 1, the holder of the token. If $a > 1$, additional reuse could be made of the packet, but the opportunity for reuse will be terminated at the original sender in this analysis.

Length of packet also presents a problem, as mentioned above. Obviously, a message which is inserted into the free slot *must* of length less than the length of the delivered message. For this analysis, the assumption is made that all packets are of equal length with the effect of varying packet lengths to be examined in subsequent research.

5.3 Objective

If *destination insertion* recaptures these packets, one expects throughput to increase and delay to decrease. The central questions are how much these measures would be affected and how feasible the implementation would be. Recent papers[6,1,4] have used similar techniques to show increases on the order of 1.5 to 2 times 100%, but these papers have not included any generalizations as to how this might apply in an arbitrary case, rather the simulation results are for specific cases. I intend to provide an analysis which will allow one to predict the degree to which a method such as this can improve performance in FDDI or token rings, and to show that a feasible strategy can be developed which does not meet the maximum, but can approach it.

5.4 Advantages

One might assume that the effect of *destination insertion* would be simply to provide for an increase in throughput at high loads and have little effect at low loads.

However, the method can be shown to improve performance in the following areas.

- Throughput will be able to sustain traffic at a much higher load
- One of the major problems with token ring networks is the access delay for obtaining the token. These extra slots will reduce average access delay to the network.
- Because of the large distance and number of nodes inherent in proposed wide area research nets, normal token ring access delays are likely prohibitive. These additional packets will reduce access delay as mentioned above and reduce the sensitivity of a token ring network such as FDDI to longer distances.

5.5 Expected Maximum Throughput Increase

For this analysis, the assumptions will be that all nodes always have at least one packet in the queue, that destination address space is uniformly distributed among the nodes, and that packets are fixed length. If all messages were destined for the neighbor, throughput could be increased by a factor of N, but this is an unlikely scenario. The result derived is a function of the number of nodes, $\mathcal{E}(n)$, which states the factor by which throughput can be expected to improve under heavily loaded conditions. For example, if utilization is currently 80% and the expected throughput increase, $\mathcal{E}(n)$, is 1.4, then utilization should be able to reach 112% under the conditions specified in the assumptions above for n nodes.

In order to determine the expected throughput for such a network, consider the traversal of a packet around the network. Assume that node n removes the token from the network and transmits a packet. The packet must be destined for one of the nodes $n-1\ldots1$. Assume that the packet is destined for node $j, n < j \leq 1$. Upon receiving the message, either

1. node j has a packet available for transmission to node k where $j < k \leq< 1$ or $k = n$ which states that the message can be squeezed into the now available slot and removed before it passes the original sender
2. node j has a packet available for transmission to node k where $n-1 < k < j$ and it can not be squeezed without a possibility of being removed by a node which has the token (specifically node n may still be transmitting, so we assume that it is in order to guarantee viability of the slot)
   or
3. there is no packet in the queue (which this analysis is ignoring).

Define $\mathcal{E}(i)$ to be the expected increase in thoughput given that the slot has as its destination node i. Using a recursive derivation, start with $\mathcal{E}(1)$.

$$\mathcal{E}(1) = \frac{1}{N} \tag{11}$$

because the expected value of increased throughput is the probability of the message in the head of the queue being destined for node n (the original sender of the slot and the only possible node to which node 1 can send), $\frac{1}{N}$, times 1 (the number of messages it can send in the slot) plus the probability that the message at the head of the queue is for some other node, $\frac{N-1}{N}$, times 0.

The expected increase at the second node is composed of three terms

1. the probability that the message at the head of its queue is for node n times its expected increase, $\frac{1}{N}$ times 1 as above, plus
2. the probability that the message at the head of the queue is for node 1, $\frac{1}{N}$, times the expected increase which is 1 plus $\mathcal{E}(1)$ because the message is delivered at node 1 and can be reused again at node 1 with expected increase $\mathcal{E}(1)$ plus
3. the probability that the message at the head of the node is for neither node 1 nor n, $\frac{N-2}{N}$, times the expected increase which is also $\mathcal{E}(1)$ because the slot will proceed to node 1 available for reuse

Therefore,

$$\mathcal{E}(2) = \frac{1}{N} + \frac{\mathcal{E}(1)}{N} + \frac{N-2}{N} \times \mathcal{E}(1) = \frac{1}{N} + \frac{N-1}{N^2} \tag{12}$$

For arbitrary node j, the formula can be generalized to

$$\mathcal{E}(j) = \frac{1}{N} + \sum_{i=1}^{j-1} \frac{1}{N} \times (1 + \mathcal{E}(i)) + \frac{N-j}{N} \times \mathcal{E}(j-1) \tag{13}$$

$$= \frac{j}{N} + \frac{N-j+1}{N} \times \mathcal{E}(j-1) + \sum_{i=1}^{j-2} \mathcal{E}(i) \tag{14}$$

for $n < j \leq 3$
and

$$\mathcal{E}(j) = \sum_{i=1}^{j-1} \frac{1}{N} \times (1 + \mathcal{E}(i)) \tag{15}$$

for $j = n$

The first term in Equation 13 represents the expected increase if the message is for node n, the original sender. The second term represents the expected increase if the message is for a node positioned between the current node j and node 1 which is the squeezed message itself plus any expected increase once that message is delivered. The third term assumes that the slot could not be used so it is passed on to node $j-1$. In the case for $j = n$, the first term is omitted because it would never send a message to node n, itself.

5.6 Overall effectiveness

The following graph show the increase in throughput expected from a traffic placement strategy as described above. The result of interest from the above derivation is the value of $\mathcal{E}(n)$ which describes the number of expected messages delivered with each packet as it is transmitted from the node holding the token (n). Figure 7 shows $\mathcal{E}(n)$ versus n. One can observe that the effect of such a technique has a much greater effect as the number of nodes increases.

Recall that the proper interpretation of this graph is that throughput can be increased by the factor given. Results show that for a 100 node problem which operates at 90% maximum utilization, this method will increase throughput by a factor of 3 to 270%. A number of curves are provided. The curve marked *Analysis* is the result of calculating $\mathcal{E}(n)$ for various values of n as derived above. The analytical results can be compared with simulation results in the curve *Simulation*. The simulation model did not require using an FDDI model and was only modelling the passing of messages from node to node without delay statistics. Here a packet was allowed to be reused an arbitrary number of times, an impractical assumption discussed later. The final two curves in this figure show maximum utilization when one limits the number of times which a slot may be used during a cycle around the network. *Max=2* indicates that the slot may be used twice (reused once).

5.7 Additional Simulation Results

A simulation model of FDDI, written in Simscript, has been developed on Sun workstations in order to test performance issues. A modification was made to the model to allow for the incorporation of *destination insertion*. It should be noted that the full advantage of this technique can not be seen in these results because of an inconsistency in the original design of the model and the design necessary for *destination insertion*. A new model which will be used to show the full potential impact for FDDI is under development. The results shown are a conservative estimate of the effect.

A number of benefits arise from removal of the message at destination and reuse of the slot, the first two of which are investigated in this paper.

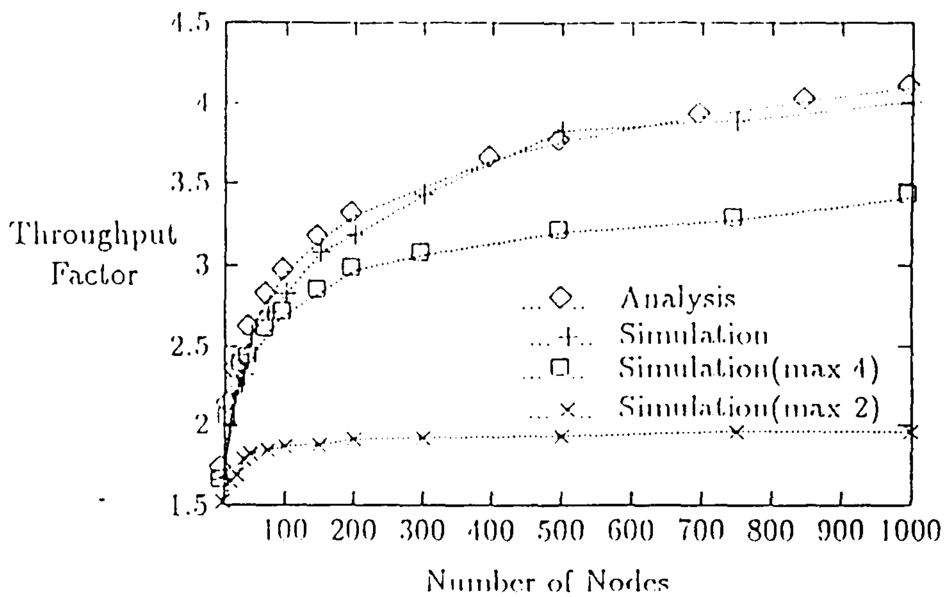

Figure 7: Expected Throughput Increase

1. As shown previously, throughput can be dramatically increased.
2. Even in scenarios where the system is not fully loaded, delay characteristics are improved.
3. The extensibility of an FDDI network is increased. One of the major limitations of extending an FDDI ring is the large propagation delays experienced as total network distance increases. These extra slots will provide additional opportunities for transmission beyond the token arrivals, decreasing access delay.
4. If synchronous traffic is required, it is possible to set $TTRT$ values at a higher value and still maintain the same average access delay. Raising the $TTRT$ will allow for higher utilization as shown previously.

The effect of access delay where load is less than 100% is shown in Figures 8 and 9. $TTRT$ is set to 5 ms, network rate(R) at 100 Mbps, packet length(P) is fixed at 5000 bits. Figure 8 shows how removal affects the 10 node case. Figure 9 incorporates the removal vs non-removal for 10, 50 and 100 nodes. Note that in every instance using the *destination insertion* technique, access delay at 100% load is comparable to access delay at very low loads. It should also be noted that these runs have not reached the assumption made in the analysis that all nodes have data in the queue; however, the effect is still significant.

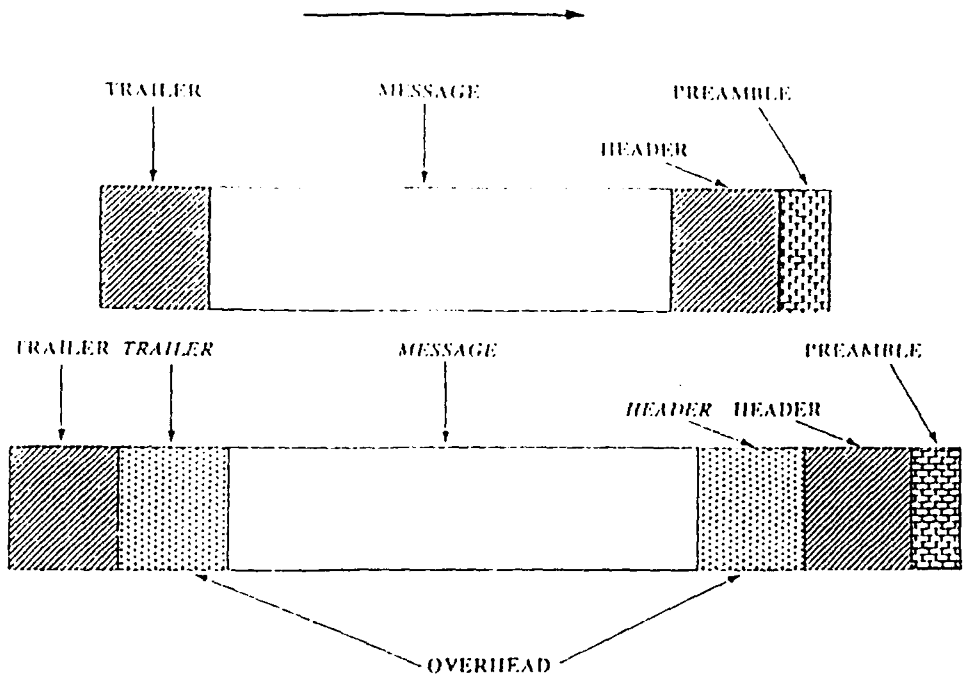

Figure 10: Packet Structure

5.8 Implementation

The implementation of this technique is not without a cost, but the cost is a nominal one. First, the receiver must be able to recognize the destination address and mark the packet as delivered. Second, the marking of the packet must be in such a manner as to allow for other nodes to determine that the packet has already reached the destination. Placing the header at the beginning of the packet allows for interpretation of the source and destination. This would make it possible for a receiver to determine if the packet had already been delivered, but the process of making the decision would be simplest if the receiver marked a 1 bit slot at the end of the header to indicate that the packet had been received. All subsequent nodes would recognize this similar to recognizing the token and transmit the data.

Token rings allow the original message to propagate back to the sender before being removed. This provides and acknowledgement mechanism that the message has been delivered and has circulated around the ring correctly. However, receipt of the original message is not required. Many protocols only require an acknowledgement of some type from the receiver. This can be done much more efficiently that using the entire packet slot for acknowledgment. If sufficient room is left at the end of the packet, the acknowledgement from the receiver can be accommodated. This would require an additional header/trailer combination for the reuse of the packet as shown in Figure 10. The number of these extra header/trailer combinations would be equal to the number of times of expected reuse of the packet. Of course, in the best case, all nodes would want to use the slot requiring $N \times (P_{trailer} + P_{header})$ overhead, an unacceptable price to pay. However, a value closer to the average number of uses of the slot could be used. For example, if on the average for a given number of nodes, the packet would be used 3 times, the overhead of 3 packet/trailer entries would be a reasonable one.

6 Conclusions

This paper has shown some of the limitations of FDDI. Those limitations are strongly a function of number of nodes and network distance. A tradeoff exists between maximum utilization and the network access as determined by the $TTRT$. Return of the token within an average specified time can be guaranteed but use of the token cannot. It has been shown that setting $TTRT$ lower reduces utilization. Reduced utilization in turn will *increase* access delay, the reverse of the desired effect.

Scalability of FDDI in the 500 node, 50Km range up to gigabit speeds is not without some proportional loss in performance. Simply increasing the transmitter speed by a factor of ten does not translate directly into being able to deliver ten times the data with similar access delays.

One method of improving the extensibility of FDDI is to remove packets at the destination and make those packets available for reuse as they continue circulating around the ring. This has been shown to have an expected increase of over twice the capacity of the network for more than 20 nodes when the network is fully loaded and to reduce access delay in cases where the network is less than fully loaded.

References

[1] Dobosiewicz and Gburzynski. Ethernet with segmented carrier. *Computer Networkin Symposium*, 72–78, April 1988.

[2] D. Dykeman and W. Bux. Analysis and tuning of the fddi media access control protocol. *IEEE Transactions on Communications*, 6(6):997–1010, July 1988.

[3] G.M. Exley and L.F. Merakos. Throughput - delay performance of interconnected csma local area networks. *IEEE Journal on Selected Area in Communications*, 1380–1390, Dec. 1987.

[4] E.C. Foudriat, K. Maly, C.M. Overstreet, S. Khanna, and F. Paterra. *A Carrier Sensed Multiple Access Protocol for High Data Rate Rings.* Technical Report, Old Dominion University, February 1990.

[5] IEEE 802.6 Working Group. Proposed standard: distributed queue dual bus man. August 7, 1989.

[6] W. Hilal and M.T. Liu. Analysis and simulation of the register-insertion protocol. *Proceedings, Computer Network Symposium*, 1982.

[7] H.S. Hinton. Architectural considerations for photonic switching networks. *IEEE Transactions on Communications*, 6(7):1209–1226, August 1988.

[8] M.J. Johnson. Proof that timing requirements of the fddi token ring protocol are satisfied. *IEEE Transactions on Communications*, 35(6):620–625, June 1987.

[9] R.E. Kahn. A national network: today's reality, tomorrow's vision, part 2. *EDUCOM Bulletin*, 14–21, Summer/Fall 1988.

[10] J.O. Limb and C. Flores. Description of fasnet - a unidirectional local-area communications network. *Bell System Technical Journal*, 61(7):1413–1440, September 1982.

[11] M. Maeda and H. Nakano. Integrated optoelectronics for optical transmission systems. *IEEE Communications*, 26(5):45–51, May 1988.

[12] K. Maly, E.C. Foudriat, D. Game, R. Mukkamala, and C.M. Overstreet. Traffic placement policies for a multi-band network. *SIGCOMM 89*, August 1989.

[13] K. Maly, C. M. Overstreet, X. Qiu, and D. Tang. Dynamic resource allocation in a metropolitan area network. *SIGCOMM Symposium*, 13–24, August 1988.

[14] M.A. Marsan and D. Roffinella. Multichannel local area network protocols. *IEEE Journal on Selected Areas in Communications*, 885–897, Nov. 1983.

[15] N.F. Maxemchuk. A variation on csma/cd that yields movable tdm slots in integrated voice/data local networks. *Bell Systems Technical Journal*, 61, no 7:1527–1550, September 1982.

[16] T. Nakagami and T. Sakurai. Optical and optoelectronic devices for optical fiber transmission systems. *IEEE Communications*, 26(1):28–33, January 1988.

[17] R.M. Newman, Z.L. Budrikis, and J.L. Jullett. The qpsx man. *IEEE Communications Magazine*, 26(4):20–28, April 1988.

[18] K. Nosu. Advanced coherent lightwave technologies. *IEEE Communications*, 26(2):15–21, February 1988.

[19] F. Ross. Fddi - a tutorial. *IEEE Communications*, 24(5):10–17, May 1986.

[20] Draft Proposed American National Standard. Fddi token ring media access control (mac) asc x3t9.5 rev. 10. February 28, 1986.

[21] F.A. Tobagi, F. Borgonovo, and L. Fratta. Expressnet: a high-performance integrated-services local area network. *IEEE Journal on Selected Areas in Comm.*, SAC-1, no. 5:898–913, November 1983.

[22] F.A. Tobagi and M. Fine. Performance of unidirectional broadcast local area networks: expressnet and fasnet. *IEEE Journal on Selected Areas in Comm.*, SAC-1, no. 5:913–925, November 1983.

[23] J. Ulm. A timed token protocol for local area network and its performance characteristics. *IEEE*, Feb. 1982.