On Token Protocols for High-speed Multiple-ring Networks

W. Dobosiewicz and P. Gburzynski
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada, T6G 2H1

Abstract

A token ring protocol which restricts the transmission rights to the station possessing the token is not a good candidate for driving a gigabit network. As the network's transmission rate becomes higher, the impact of the token transition time (which is idle) becomes more pronounced and the effective throughput of the network becomes a smaller and smaller fraction of the nominal channel capacity. We present a family of token passing protocols for ring networks which are devoid of this unpleasant property. The protocols are simple and inexpensive, yet they possess a number of advantageous properties as high capacity (similar to METARING *and independent of the propagation length of the ring), fairness, and natural accommodation of synchronous and isochronous traffic.*

1 Introduction

To focus our discussion on the logical aspects of the investigated protocols, we will use the *normalized propagation delay* (expressed in bits) to measure time, distance, and packet length [9]. We also abstract from coding issues, which may otherwise obscure the discussion, and assume that signals transmitted through the ring consist of bits. In particular, an FDDI symbol (encoded using the 4B5B code—[6, 7]) is composed of 4 bits, not 5.

Let L denote the round-trip propagation delay of the ring (expressed in bits) and l_p be the average length of a packet transmitted through the network. The ratio L/l_p is commonly denoted by a. A large value of L, and consequently a, may result from a very high transmission rate of the network, a very long geographic length of the ring, or a combination of these two properties. The performance of many protocols (not only ring protocols) deteriorates when a becomes significantly bigger than one. For example, consider the following simple formula expressing the maximum throughput of FDDI:

$$T_{FDDI} = \frac{\Sigma THT}{\Sigma THT + L} \qquad (1)$$

where ΣTHT represents the sum of the token holding times at all stations (and can be viewed as a representative of the packet length l_p). Although one may naively try to increase the maximum throughput of FDDI by increasing ΣTHT, this approach has obvious limitations: it assumes that each station has sufficiently many ready packets to fill its token holding window; moreover, it increases the average ring access time.

Ideally, the formula expressing the maximum throughput of a ring network should be independent of L (or at least its value should not be inversely proportional to L). Protocols with this property are called *capacity-1* protocols: they are able to utilize a fixed portion of the channel bandwidth, irrespective of the propagation length of the network. To demonstrate that designing protocols with this property is not trivial let us consider METARING [2] which is one of the best protocols for ring networks known to the authors. The maximum normalized throughput of "pure" METARING (without the SAT mechanism) does not depend on L. Assuming a uniform distribution of traffic, this throughput is equal to 8 (4 per each of the two rings), which is the upper bound on the throughput of any network based on two counter-rotating rings. Unfortunately, in its pure version, METARING is starvation prone.

To eliminate starvation in METARING, a special mechanism (SAT) is introduced imposing a limit k on the number of packets that a station can transmit before it is obliged to yield the bandwidth to other backlogged stations. With this mechanism, the maximum throughput of METARING is given by:

$$T_{META} = min\left(\frac{2Nk}{L}, 8\right) \qquad (2)$$

where N is the number of stations in the network

0-8186-3670-X/93 $03.00 © 1993 IEEE

and k is the above-mentioned limit. Clearly, this formula depends on L in the "unpleasant" way. Although one may observe that by increasing k, the impact of L can be formally eliminated, the protocol's fairness and its ability to respond to varying load patterns is impaired in direct proportion to k. When $k = \infty$, the SAT mechanism is turned off and the network becomes starvation-prone. On the other hand, any criticism of METARING in this respect should be put into a proper perspective: at the level of throughput that its main competitors (today) can handle, METARING guarantees a fair and starvation-free access to the media; however, it happens at a much greater hardware expense per station. It would be nice to be able to produce a formula for this tradeoff, but in the fast-pace world of electronics, it is totally impossible.

The above example is rather typical. The tradeoff between fairness and maximum effective throughput (and, implicitly, cost) becomes clearly visible for higher values of L. Intuitively, fairness is often achieved by a direct or indirect form of negotiating medium access across the network. Such negotiations incur access delays proportional to L. If stations are not allowed to use the bandwidth before it has been negotiated, the maximum effective throughput must deteriorate with increasing L. On the other hand, stations trying to take advantage of idle negotiation periods—to improve throughput—run the risk of preempting other needy stations in an uncontrolled (unnegotiated) way.

In this paper, we propose a family of protocols for ring networks which come close to possessing all the properties of an ideal protocol. We focus on the *capacity-1* property, fairness, and accommodation of synchronous traffic. These properties are most important from the viewpoint of gigabit applications. The proposed protocols are also simple, flexible, and predictable.

2 Multiple weak tokens

2.1 Protocol rules

Imagine a ring network operating in a slotted manner. The slots are inserted into the ring upon initialization and since then they circulate indefinitely. The slot header includes two special bits. The *full* bit indicates whether the slot carries a packet[1] (the *full* bit is 1) or is empty and can be used to insert a packet (*full*= 0). The other bit is used to pass the token. Normally, the *token* bit is set to 1 by the by the station releasing the slot, unless the station decides to pass the token to its successor, in which case it clears the token bit.

There are multiple tokens circulating in the ring. The protocol family is called *MWT*, for *Multiple Weak Tokens*. The tokens are *weak* because, in contrast to *Strong Token* protocols (e.g., FDDI), token possession is not a necessary condition for transmission.

If each token is always held for the same amount of time at every station, the distance between the tokens (measured in slots) is fixed. If this distance is small, it may happen that a station holding a token receives another token. The second token is then held by the station independently of the first one—by the prescribed amount of time, so that the two tokens depart from the station separated by the same interval of slots as upon their arrival. Note that it is not absolutely necessary that all stations hold all tokens by the same amount of time. Different stations may use different token holding intervals as long as the same interval is applied to all tokens held by a given station. For simplicity, we will assume that all stations observe the same holding time; it should be clear how this assumption can be relaxed.

A station holding the token is responsible for cleaning, i.e., stripping all segments arriving from upstream. This is done by unconditionally clearing the *full* bits in the headers of all incoming slots.

Every station counts the time (in slots) elapsed since the moment the station last released a token. To accomplish this, the station maintains a counter called TT_i, where i is the station index. This counter is set to zero at the moment when the station passes the token (at the beginning of the slot in which the token is passed) and then incremented by 1 whenever the station receives a new slot. There is only one simple transmission rule. A station i ready to transmit waits for the moment when the following condition is satisfied:

$$TT_i \geq (D-1) \times THT \tag{3}$$

where D is the number of forward "hops" separating the recipient station from the sender.[2] Starting from this moment, the station transmits the segment in the first free slot. Condition (3) ensures that the transmitted segment will not be absorbed by a token holding station, before it has reached the destination.

Assume that a station has a segment to transmit to a destination located D hops down the ring. The station waits until condition (3) is fulfilled. This requires

[1] Following the terminology of DQDB [5], we will call it a segment.

[2] Immediate neighbors are separated by one hop.

a gap of at least $(D-1) \times THT$ slots between the last token departure from the station and the departure of the next token. Note that possession of a token is neither a sufficient nor a necessary condition for transmission. Consider a dual counter-rotating ring configuration with N stations. The maximum number of hops to be traveled by a segment is $D_{max} = \lfloor N/2 \rfloor$. The two rings are independent and following the simple ring selection operation, the fate of a segment is confined to the selected ring.[3] In fact, the sole purpose of the other ring (besides providing additional bandwidth) is to reduce D_{max}. The minimum condition to make such a network operable is the existence of at least one pair of adjacent tokens separated by at least $(D_{max} - 1) \times THT$ slots. Otherwise, it would never be possible to send a segment to the most distant destination.

Although a station holding a token may not be able to transmit a packet to a distant receiver, it still enjoys the privilege of acquiring empty slots for transmission. This, however, brings the issue of **fairness**, which we will define here in the following way: *a medium access strategy is fair, if it does not discriminate against any destinations.*[4]

The idea behind the multiple tokens is to provide a mechanism for spatial reuse: segments should be removed before they reach their senders, preferably, immediately after they have reached their destinations. This brings us to the following postulates:

- The maximum space between two consecutive tokens should not be greater than $(D_{max} - 1) \times THT$. Larger token spacing brings no new transmission opportunities, but reduces the number of tokens which has a detrimental impact on the maximum throughput achievable by the network (see below).

- To maximize throughput, a station with several segments awaiting transmission should select a segment addressed to the farthest destination reachable at the current moment. This way, the amount of bandwidth wasted by the segment after is has passed its recipient will be minimized. Ideally, if each station has a variety of segments to choose from, the protocol may emulate destination cleaning without buffering the segments at the destinations.

- Under heavy traffic (all stations constantly backlogged), all transmissions are done by token holding stations. To avoid starvation, we postulate the following informal condition (which will subsequently be formalized):

> For each D, $1 \leq D \leq D_{max}$, and some positive constant $\mathcal{K}$, there should exist exactly $\mathcal{K}$ tokens T_D such that the maximum transmission distance for a station holding a token T_D is exactly D.

If this condition is not fulfilled, then, when the traffic is heavy, packets to some destinations may be starved. The requirement that there are exactly $\mathcal{K}$ tokens of each given reach is meant to ensure fairness; it is unlikely that it can be met completely (since L and N are given in advance), so we settle for a slightly weaker requirement: the number of situations in which a station can reach a destination located D hops down the ring should be **approximately** the same for all D, $1 \leq D \leq D_{max}$.

The maximum throughput of MWT is very simple to calculate. Under heavy load, all transmissions are performed by token holding stations. Assuming that every station has a constant backlog of segments to transmit, every station transmits through its entire THT interval, and only then. Thus, the maximum throughput of MWT is given by the following formula:

$$T_{MWT} = M \times \frac{TRT - L}{TRT} \qquad (4)$$

where M is the combined number of tokens in both counter-rotating rings. Although the second factor decreases with increasing L (this part describes the throughput of a strong token protocol), we will shortly see that the number of tokens M (constrained by our postulates) is proportional to L. Thus, MWT is a *capacity-1* protocol.

2.2 Token allocation

We assume that we are given three parameters: the ring length L in slots, the number of stations N, and the value of THT (token holding time) per station and per token. The last parameter can be flexible, however, in most cases its value is constrained by the application profile. It is always reasonable to keep the token holding time small. Based on these parameters we would like to determine the sequence $V_0, \ldots, V_{M-1}$ of integer numbers describing the configuration of tokens to be inserted into the ring upon initialization.

[3] In contrast to METARING, where backward feedback information for one ring travels along the opposite ring.

[4] This is not the only way to define fairness; see [4]. Note that in our case "source" fairness is guaranteed by token circulation.

V_i is the interval (in slots) between a pair of adjacent tokens. We have:

$$\sum_{i=0}^{M-1} V_i = TRT = L + N \times THT \qquad (5)$$

The problem of determining the best configuration of tokens for a particular network can be solved in two steps. First, we determine the collection of values $V_0, \ldots, V_{M-1}$ treated as a set, i.e., without assuming any specific ordering of these values. Note that from the point of view of fairness under heavy load, the permutation of tokens is irrelevant. The only important point is that all distances to destinations are included in the collection of token intervals in approximately the same multitude. By permuting the token intervals we influence the medium access time for light load; this is a separate problem.

Assume that the load is heavy and a station S just receives a token T_i. Let the interval between T_{i-1} (the previous token seen by S) and T_i be V_i. Since the traffic is heavy, S can only transmit while it holds the token—for THT slots. To be able to transmit to a station located at distance D, the number of slots elapsed since the departure of T_{i-1} must be at least $(D-1) \times THT$ (see condition (3)). Thus, in the first slot of its token holding interval, S can transmit at distance $D_i(0) = \lfloor \frac{V_i - THT}{THT} \rfloor + 1$. In general, during the j-th slot of the token holding interval, the maximum distance at which S can transmit is given by the following formula:

$$D_i(j) = \lfloor \frac{V_i - THT + j}{THT} \rfloor + 1$$

While holding token T_i, station S may transmit THT slots. These slots can be grouped in sets based on the distance they will travel before reaching token T_{i-1}. Let $K_D(V_i)$ be defined as the set of all slots that will travel a distance D before reaching token T_{i-1}, for $1 \leq D \leq D_{max}$. Thus,

$$K_D(V_i) = \{j \in [0, THT-1] : D_i(j) = D\}$$

Note that $K_D(V_i)$ is empty if D never happens to be the maximum transmission distance while the station is holding token T_i. If $THT = 1$, there is exactly one nonempty set K_D for a given V_i.

Considering all the tokens,

$$r_D = \sum_{i=0}^{M-1} \overline{K_D(V_i)}$$

gives the total number of slots during which a token holding station can transmit at the maximum distance D. We define the unfairness of a given token allocation scheme as:

$$U = \max |r_A - r_B|, \quad 1 \leq A, B \leq D_{max} \qquad (6)$$

The problem of ensuring fairness is equivalent to minimizing U. Although this integer optimization problem is hard to solve by a closed formula (due to its discrete nature), it can be successfully attacked by approximate methods. We have devised a genetic algorithm for finding solutions with small U, which operates along the lines described below.

A starting configuration of token intervals is built according to the following scheme (described in C):

```
D = D_max; i = 0; left = TRT;
while (left != 0) {
    V_i = (D - 1) × THT + ⌊THT/2⌋;
    if (V_i < THT) V_i = THT;
    if (V_i > left || left - V_i < THT) V_i = left;
    left = left - V_i;
    i = i + 1;
    D = D - 1;
    if (D == 0) D = D_max;
}
M = i;
```

The initialization procedure continues until the entire TRT interval is exhausted (see equation 5). In each turn, it services one distance between 1 and D_{max}: it tries to build a token interval for which the given distance is the longest transmission distance in the middle slot of the token holding interval. All distances are served cyclically, starting from D_{max} down to 1, then again proceeding from D_{max}, and so on.

Then, the algorithm calculates the unfairness of the initial configuration and performs a number of iterations trying to improve this unfairness by adjusting boundaries between adjacent intervals. The intervals are permuted in a randomized way by grouping together the intervals whose adjustments resulted in the biggest improvement. The algorithm may decide to add a new interval (by inserting a token in the middle of an existing interval) or to combine two intervals into one. In all cases, the resulting number of tokens M ends up very close to the initial number produced by the initialization procedure listed above. To estimate this number, note that the average length of a token interval after initialization is of order $(D_{max}/2 - 1) \times THT + THT/2$ which translates into $(N-2) \times THT/4$, assuming the number of stations N is even. The total number of tokens can be estimated by dividing $TRT = L + N \times THT$ by the average length of a token interval which yields:

$$M \approx \frac{4L}{(N-2) \times THT} + \frac{N}{N-2} > \frac{4L}{TRT - L} + 1 \quad (7)$$

In combination with formula 4 and assuming two symmetric counter-rotating rings, this gives us the following estimate on the maximum throughput achieved by MWT:

$$T_{MWT} > \frac{8}{1 + \frac{TRT-L}{L}} \quad (8)$$

Somewhat paradoxically, and contrary to most medium access protocols, the maximum throughput of MWT tends to improve with increasing L. Asymptotically, it approaches 8 which equals the throughput of pure, dual-ring, buffer-insertion METARING without *SAT*. MWT achieves this throughput without explicit destination cleaning and without buffering segments at intermediate stations. Moreover, MWT is fair and starvation-free without sacrificing any portion of the bandwidth to implement this feature. It is also well synchronized (tokens arrive at stations at very regular intervals) which makes it well suited for synchronous and isochronous applications.

Let us now devote some attention to the medium access delay under light traffic conditions. Suppose we are given an allocation of token intervals $V_0, \ldots, V_{M-1}$. This time the ordering of these intervals is important so we assume that the tokens circulate in the ring in the listed order. Consider a random station S and the slots that visit S between two consecutive arrivals of a token (say, T_0) to S. There are TRT such slots; let them be numbered from 0 to $TRT-1$, according to the order in which they visit S. Consider a given distance D, $1 \leq D \leq D_{max}$. Let $t_0^D, \ldots, t_{N_D-1}^D$ be the ordered sequence of slot numbers, induced by $V_0, \ldots, V_{M-1}$, identifying the slots within which a transmission at distance D is possible. These are all the slots satisfying condition (3). Assume that the network is idle and S gets a segment addressed D stations down the ring. The probability that the packet arrives at S while the station is between slots t_i^D and t_{i+1}^D is given by the following formula:

$$P_i^D = \frac{t_{i+1}^D - t_i^D}{TRT}$$

To avoid discussing the special case when S is between $t_{N_D-1}^D$ and t_0^D, we put $t_{N_D}^D = t_0^D + TRT$. Then, the expected waiting time is $(t_{i+1}^D - t_i^D)/2$. Consequently, the expected access delay for transmission at distance D is:

$$A_D = \sum_{i=0}^{N_D-1} \frac{(t_{i+1}^D - t_i^D)^2}{2TRT}$$

If all destinations are equally likely, the expected global access delay under light load is obtained by averaging A_D over all D, i.e.:

$$A_{MWT} = \frac{\sum_{D=1}^{D_{max}} A_D}{D_{max}} \quad (9)$$

For a biased distribution of destinations, the contribution of particular A_D's in formula 9 should be weighted properly.

The problem of minimizing the access delay under light load boils down to finding the permutation of the token intervals $V_0, \ldots, V_{M-1}$ that minimizes A_{MWT}. Again, we are generally able to find good solutions to this problem with a genetic algorithm that locates at random permutations with a "reasonable" A_{MWT} and then tries to improve them in a methodological way.

While the above discussion was based on using two physical counter-rotating rings, it may be extended to any number of rings. For example, with $2\mathcal{R}$ rings ($\mathcal{R}$ clockwise and $\mathcal{R}$ counterclockwise), the same multiple-token protocol works correctly and yields a maximum throughput exceeding:

$$\frac{8\mathcal{R}}{1 + \frac{TRT-L}{L}}$$

If the tokens are staggered appropriately,[5] adding more rings will reduce the average access delay under light load. Although building a network of a large number of physical rings may be an uninteresting proposition, the above extension also applies to *logical rings*, e.g. WDM-based networks [10].

2.3 Strengths and Weaknesses of MWTP

Since the Multiple Weak Token protocol may be seen as an alternative to METARING, it is natural to compare the two. In comparison to METARING, the use of multiple tokens has the following properties:

Advantages

- No need for an insertion buffer.
- A much simpler mechanism for spacial reuse, one that does not require fast reaction of stations.
- Starvation free.
- A simple mechanism guaranteeing each station its share of the total bandwidth, when needed and only then.

[5] Similar to disk striping.

- Lesser vulnerability to station failures. Packets cannot be "orphaned."
- Predictability of network events, which allows automatic healing of failures on the fly.

Disadvantages

- Unsuitable for small networks, especially small networks with a large number of stations.
- Slightly smaller maximum throughput for uniform traffic.
- A greater average access delay, especially for light loads.
- Problems with a heavily biased traffic pattern in which the receiver is located exactly $N/2$ stations away from the sender. The protocol is not able to assign more than $2/N$ of the total bandwidth to such a traffic pattern, unless the presence of such pattern is known in advance (then, a different token allocation scheme will give such pattern no less bandwidth than METARING).

Whether the net balance is positive depends on the applications using the network.

3 Simulated environment

We used simulation to compare the performance of MWT, FDDI, and METARING. The number of messages transmitted during a single experiment varied as a function of the network size, the offered load, and the token rotation time; its range was between 200000 and 2000000 messages.

Two lengths of a ring were considered: 10^5 bits and 10^6 bits. Assuming a 200 km ring, the two propagation lengths represent transmission rates of $100Mb/s$ and $1Gb/s$. Note that FDDI was designed for smaller lengths and is not expected to work efficiently beyond the 10^5 bit range.[6]

The number of stations was the same for all networks and equal 33. Since both METARING and MWT assign packets to rings based on the distance between the sender and the receiver, it is convenient to use an odd number of stations (stations are equidistant in the experiments).

[6]The comparison to FDDI is not intended to belittle this protocol, but rather to show that the adoption of the "weak token" concept allows projecting its use into the gigabit range.

The traffic was uniform in the sense that the probability of every pair (*sender*, *receiver*) was the same. For simplicity, networks were assumed to be **slotted** and the slot format of DQDB was assumed: each slot consisted of a 384-bit payload and 40-bit header. This assumption, while inconsistent with the formal definitions of FDDI and METARING, does not alter the validity of the results: first, all the protocols were affected in the same way; second, none of the protocols takes advantage of a fixed packet size. Additionally, FDDI was assumed to operate on both counterrotating rings, instead of using one of them as a standby ring. For all the protocols, messages were assigned to transmitter queues at the moment of arrival and there were not moved from queue to queue.

In MWT, THT was the same for each station and equal to 1 slot. In FDDI, the target token rotation times $TTRT$ were taken to be:

- 1.5×10^6 bits for the 10^5 bit rings, which is equivalent to 15 ms.
- 2.4×10^6 bits and 15×10^6 bits for the 10^6 bit rings. These values are equivalent to 24 ms and 150 ms, respectively.

The message length was fixed and equal to the segment payload size.

In METARING, the SAT mechanism was not used (ie, k was assumed to be ∞).

4 Performance measures

We measured the mean message access delay (A in the figures), defined as the mean time elapsing from the moment when message is enqueued for transmission and the moment when its first bit is successfully transmitted. The delay is expressed in bits. Note that a meaningful comparison of the access delay of FDDI and MWT with that of METARING should take into account that in METARING, packets are buffered at each station (at least their headers), which, on average, delays the message delivery by at least 340 bits in a 33 stations network (this delay is not included in the performance measurements).

The maximum effective throughput of FDDI operating on a single ring is given in 1; this value should be doubled for a two-ring version. Likewise, the maximum throughput of METARING is given in 2. Since we assumed $k = \infty$, this maximum equals 8. The maximum throughput of MWT depends on the number of tokens used and the values of L and $TTRT$, as discussed in a previous section. In theory, a maximum of 8 could be reached (equation 8).

5 Comparison

Figures 1 and 2 show the message access delay versus throughput. METARING performs best, which is not surprising, since it uses more powerful hardware: an insertion buffer (for destination reuse of slots). In these figures, "`mwt,r`" represents a random permutation of the best token allocation obtained using the algorithm of section 2.2.3, while "`mwt,m'`" represents the permutation that is supposed to minimize delay under light load.

Figure 3 shows how the delay incurred by MWT depends on the number of stations. Note the strange behavior of the 129-stations network near saturation point; the number of stations is too large for a good token allocation in a ring of this size and packets to be sent to distant receivers are starved.

Since in MWT, one can allocate tokens in different ways, figure 4 demonstrates the performance of a sample of token allocations. In many applications, throughput is not the only important measure to be optimized. Many applications place a greater emphasis on the regularity of transmission opportunities (for isochronous traffic) or on a greater predictability (for failure recovery). In the case of MWT, the tradeoff between throughput and other considerations is particularly eminent. Figure 4 compares four different token allocation schemes in a 10^5 bit network with 33 stations: the scheme that maximizes fairness under heavy traffic conditions (32 tokens—see figure 1), 17 equally spaced tokens with the maximum reach of 16 stations each, 22 tokens with alternating reach of 8 and 16 stations, and the degenerate version of the protocol with a single token (denoted by `swt`). One can clearly see a tradeoff between the maximum throughput achievable by the network and the access delay for light load. It is conceivable to make the number of tokens vary in response to changing load patterns.

6 Summary

We presented a MAC-level protocol for a ring network. The protocol is based on passing multiple tokens. Its performance is similar to that of METARING, but it requires less hardware support, supports isochronous applications better and is self-healing after either a loss of station or a loss of token.

The protocol is of particular use in high-speed networks, since its performance actually improves with the increasing transmission rate or the size of the network.

References

[1] J. Chen, I. Cidon, and Y. Ofek. A local fairness algorithm for gigabit lans/mans with spatial reuse. *IBM Technical Report*, RC 18114, 1992.

[2] I. Cidon and Y. Ofek. A full-duplex ring with fairness and spatial reuse. In *IEEE INFOCOM'90*, pages 969–981, 1990.

[3] R. Cohen and A. Segall. Multiple logical token rings in a single high-speed ring. *Technion Technical Report #738*, 1992.

[4] W. Dobosiewicz, P. Gburzyński, and V. Maciejewski. A classification of fairness measures for local and metropolitan area networks. *Computer Communications*, 15:295–304, 1992.

[5] IEEE Project 802.6, Proposed Standard, Distributed Queue Dual Bus Metropolitan Area Network. Doc. No. P802.6/D15, Oct. 1990.

[6] Fiber Distributed Data Interface (FDDI) – Token Ring Media Access Control (MAC). American National Standard for Information Systems, Doc. No. X3, 139–1987, Nov. 1987.

[7] Fiber distributed data interface, System Level Description. Digital Equipment Corporation, June 1990.

[8] A. Kamal. On the use of multiple tokens on ring networks. In *Proceedings of IEEE INFOCOM'90*, pages 15–22, San Francisco, CA, June 1990.

[9] L. Kleinrock. *Queuing Systems*. John Wiley & Sons, Inc., 1975.

[10] B. Mukherjee. Wdm-based local lightwave networks part I: Single-hop systems. *IEEE Network*, 6(3):12–27, May 1992.

[11] A. Pach, S. Palazzo, and D. Panno. Improving DQDB throughput by a slot preuse technique. In *ICC'92*, Chicago, USA, June 1992.

[12] H. Wu, Y. Ofek, and K. Sohraby. Integration of synchronous and asynchronous traffic on the metaring architecture and its analysis. *IBM Technical Report*. RC 17718, 1992.

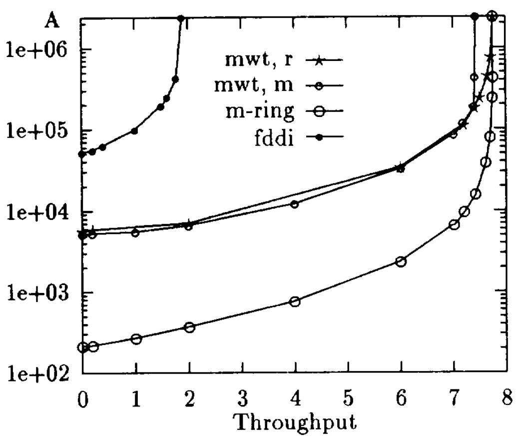

Figure 1: MWT versus METARING and FDDI, 100kb ring, 33 stations.

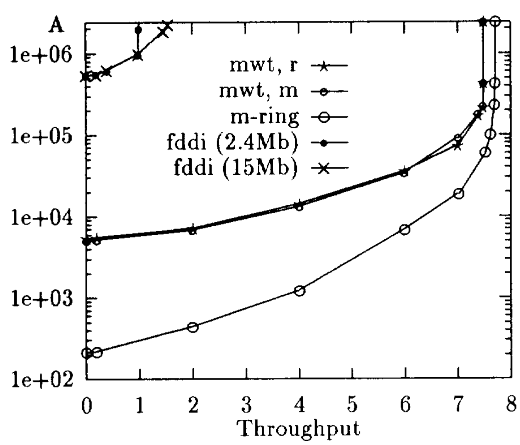

Figure 2: MWT versus METARING and FDDI, 1mb ring, 33 stations.

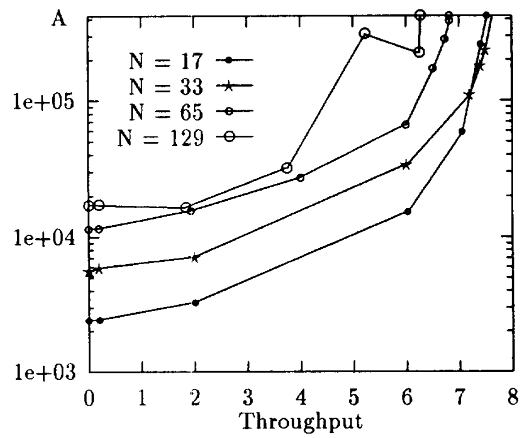

Figure 3: MWT (100kb) for different numbers of stations.

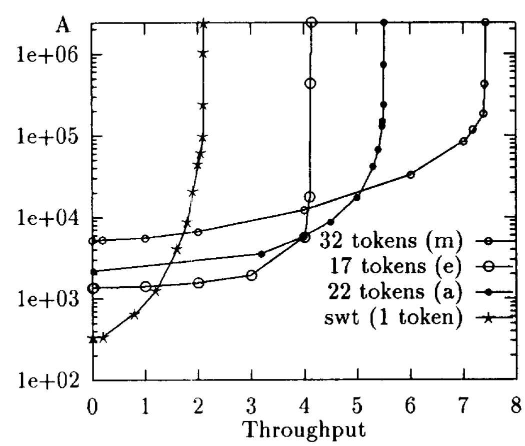

Figure 4: MWT/SWT for different token allocation strategies.