An alternative to FDDI: DPMA and the pretzel ring

Włodek Dobosiewicz, *Member, IEEE* and Paweł Gburzyński, *Member, IEEE*

Abstract—**This paper proposes an efficient, fair, and simple *capacity-1* protocol replacing FDDI.**

In the recent years, two network protocols gained predominance in very fast Local and in Metropolitan Area Networks: FDDI and DQDB. The FDDI ring–protocol combination is well-suited for large local ("campus") area networks. However, for larger networks, FDDI users suffer excessive access delays, even under very light traffic. Moreover, FDDI achieves full capacity only in special circumstances, and heavy load is not a sufficient condition for achieving it.

We propose a network topology that is very similar to the dual ring and a token-passing protocol that yield a true *capacity-1* network even for modest values of the TTRT. Its additional properties include total fairness and very small access delays under light load. This makes the proposed protocol a reasonable option to use in combination with the dual counter-rotating ring.

The proposed network should not be difficult to implement using standard FDDI components.

I. INTRODUCTION

This paper presents a network topology and protocol that combine some of the best features of FDDI [1] and DQDB [2], while not possessing any of their major weaknesses. This topology/protocol combination is best suited for a Metropolitan Area Network (MAN) or a very fast Campus Area Network.

Following tradition, we will use the *transmission bit* as a unit of time—its value is linked to the `second` by the transmission rate, e.g., a transmission rate of `1 Gb/second` implies that `1 transmission bit` $= 10^{-9}$ `second`. Likewise, the *propagation bit* will be used as a unit of distance. In this case, its value is linked to the *metre* by the propagation speed and the value of the *transmission bit*; e.g., a propagation speed of 2×10^8 `m/second` and `1 transmission bit` $= 10^{-9}$ `second` imply that `1 propagation bit` $= 0.2$ `metre`. The two new units will be referred to as *bit*. Note that we abstract from coding issues, which may otherwise obscure the discussion, as FDDI uses a 4B5B code. We use information bits, so that an FDDI symbol is made of 4 bits, not 5.

We concern ourselves with networks that are either very fast or very long—these two properties being equivalent from the point of view of the network behaviour.

Let L denote the maximum propagation delay between a pair of stations and l_p denote the average packet length (also in bits); clearly, l_p is independent of L. The ratio of L/l_p is commonly denoted by a. A large value of L (and thus a) can result from one of two[1] properties of a network (or their combination):

- a very long network (in *metres*),
- a very high transmission speed, i.e., a small ratio of the *propagation bit* to the *metre*.

The majority of the known protocols for ring or bus LANs work efficiently only if a is not much greater than 1; their performance degrades promptly with increasing a, for a variety of reasons:

- In collision protocols, packets must be inflated to be no shorter than $2L$.
- Most collision-free protocols organise packet traffic into rounds, the length of a round must include periods of silence that sum up to a value proportional to L (FDDI is an example).
- Slotted protocols often introduce delays proportional to L in order to enforce fairness [3].

Owing to advances in technology, "*Medium–a*" networks are currently available and "*Big–a*" networks are technically feasible.[2] Thus, there is a demand for protocols that perform adequately for very large a, i.e., are capable of using a fixed fraction of the capacity of the medium regardless of the value of a. We call such protocols **capacity–1 protocols**.

A number of *capacity–1 protocols* have been proposed in the literature ([2], [4], [5], [6]). This study focuses on FDDI, which is not a *capacity–1* protocol. Our main goal is to propose a *capacity–1* variation that retains the best features of FDDI, preferably using the same components as FDDI.

II. NETWORK TOPOLOGY

Fibre optics brought the *unidirectional medium* to network architectures. In this paper, we investigate single unidirectional media and dual unidirectional media.

A. Single unidirectional ring

The simplest ring topology is the single unidirectional ring, as shown in figure 1.

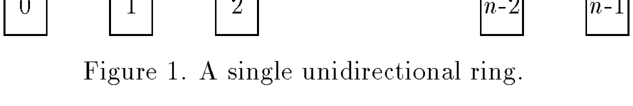

Figure 1. A single unidirectional ring.

Paper approved by Peter Martini, the Editor for Network Architecture:LANs/MANs of the IEEE Communication Society.
Manuscript received July 30, 1991; revised November 12, 1991, December 17, 1991.

The authors are with the Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2H1.

---

[1] We do not concern ourselves with the third possibility: a small propagation speed, which is typical of acoustic or horse-back networks.

[2] *Medium* and *Big* are very vague—they represent values of a of the order of 10 and 100^{+}, respectively.

In this topology, each station has one transmitter and one receiver. A suitable MAC–level protocol must be used to control the flow of packets in such a network; this protocol has two main duties:

- to determine when a station is allowed to transmit.
- to identify which station is responsible for removing packets from the ring (thus turning the ring into a *de facto* bus). This responsibility must move from station to station (as in TOKEN RING protocols); otherwise, not all the stations are reachable pair-wise. An alternative is to make stations responsible for removing parts of the traffic (each packet is removed by its sender or by its receiver)—as in ERASURE RING and INSERTION RING protocols.

Formally, a single ring can be used in combination with FDDI, although the commercial network is based on two counter-rotating rings.

B. Dual counter-rotating ring

The dual counter-rotating ring is shown in figure 2.

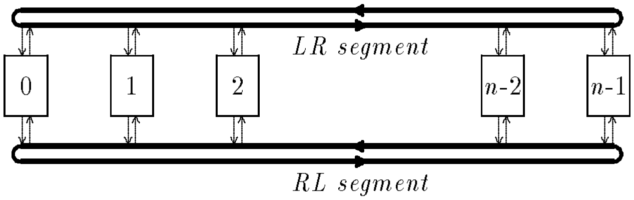

Figure 2. A dual counter-rotating ring.

Each station connected to the dual ring has two receivers (one per ring) and either one or two transmitters. If it has one transmitter, this transmitter is connected to both rings; otherwise, each transmitter is connected to one ring only.

As for the single ring, a suitable MAC–level protocol must be used to determine when a station is permitted to transmit and also to identify which station is responsible for removing packets from each ring (thus turning the rings into busses). A different station may have this responsibility for each ring. This responsibility may either be assigned to one or more stations permanently—as in DQDB—in which case there is no reason for calling the medium a ring as opposed to a bus; it may also move from station to station dynamically, or be distributed (the options are the same as for the single ring).

The dual ring is the recommended configuration for FDDI networks. One ring is used in normal transmissions (hence FDDI is commonly thought of as a single unidirectional ring network); the other is a standby ring to be used in case of component failure.

C. The PRETZEL RING

We propose a different network topology presented in figure 3. It is obtained from the dual ring by cutting both rings in the same spot and splicing each to the other ring. Unlike the dual counter-rotating ring, both components of the PRETZEL RING have the same orientation.

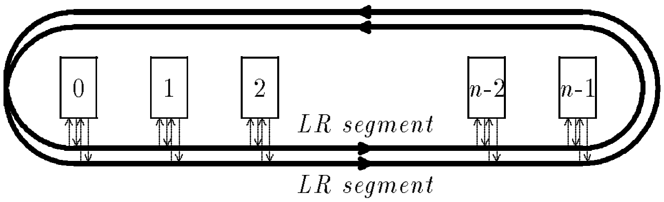

Figure 3. The PRETZEL RING.

All the other properties of the PRETZEL RING are the same as for the dual counter-rotating ring. This includes the need for a MAC–level protocol, which has the same duties as a protocol for the dual counter-rotating ring.

III. PROTOCOLS

This section describes FDDI and a new protocol that controls the flow of packets on a PRETZEL RING network.

A. FDDI

FDDI is a non-slotted protocol. Every packet must be preceded by a **preamble** of sufficient length.[3] This preamble is needed to synchronise the clock of the receiver with the transmission speed of the sender. The length of a packet has to be bounded for implementation reasons—typically to 36000 bits.

The FDDI protocol operates on one unidirectional ring (possibly using another standby ring for failure recovery). It has two responsibilities:

- FDDI determines when a station is allowed to transmit, using **token-passing**. A **token** is a special short packet that can be distinguished from all other packets. When a station receives a token packet, it destroys it; consequently, this station will assume that it is in possession of the token, which gives it special status. A station that is in possession of the token has the right to transmit packets; no other station has this right. When the token-holding station finishes transmitting, it generates a new token packet and transmits it.
- FDDI also determines how packets are removed from the ring. The token-holding station has the obligation to remove from the ring all the traffic that reaches it (equivalent to disconnecting the ring). Besides that, every station is responsible for stripping from the ring all the packets that it created. As this can only be done after repeating the header (the source address being the last field in it), a remnant of the packet is left in the ring until it reaches a token-holding station.

In order to avoid monopolising the network by one station, an upper bound is imposed on the amount of time that a station can be in possession of the token: this time may vary from station to station, yielding a weak priority mechanism. From these upper bounds, the stations may derive the maximum amount of time that may elapse between two consecutive token captures by each station; this time is called the *Target Token Rotation Time*, or TTRT. The TTRT is equal to the propagation delay of the ring plus

---

[3] Typically 64 bits, but the standard requires that stations recognise it correctly if it is at least 48 bits long.

the upper bound on the sum of token holding times for all the stations. Note that when a station receives an *early token*, which implies that not all the other stations used their full share, it may increase its own share accordingly.

B. DPMA

DPMA (*Distributed Pretzel Multiple Access*) is a token-passing protocol for the PRETZEL RING topology. In contrast to FDDI, a station doesn't have to acquire the token to start a packet transmission. Therefore, DPMA incurs zero medium access delay when the traffic is low. Like FDDI, DPMA is unslotted; thus, a whole packet can be transmitted as one entity, provided that its length does not exceed a fixed maximum length.

Each station has two receivers and one transmitter and is connected to the ring in two locations. Each connection consists of a receiver/transmitter tap. As there is only one transmitter per station, it is connected to the taps via a switch, so that, at any given moment, it can serve only one tap. At any given time, one of the two locations has a priority over the other. This higher-priority location is labelled **primary** and the other location is called the **secondary** location. The labelling of the two locations changes dynamically; however, irrespective of which location is currently primary and which secondary, the downstream path from the primary location to the secondary location includes one tap of each station. Let the distance between the two locations of a station be L; note that this distance is the same for every station.

Like every MAC–level protocol for a ring-like topology, DPMA has two main responsibilities:

- To determine when a station is allowed to transmit and—since there are two possible transmitting locations—from which location.
- To determine how the ring is disconnected and how packets are removed from the network.

A token-passing mechanism identical to FDDI's is used to single out the station that disconnects the ring. The station that holds the token disconnects the dual ring in one place—in its primary location. A station i must hold the token for an amount of time equal to the value of THT_i, a constant associated with the station (as in FDDI, priorities may be implemented by assigning different values of THT to different stations). Unlike FDDI, stations are not expected to release the token early if they have no packets to transmit; they are supposed to hold it for the whole amount of time assigned to them.

As each station holds the token for a fixed amount of time, the time that elapses between two consecutive token captures by the same station is constant and the same for all stations. Preserving the terminology of FDDI, we will denote this time by TTRT which stands for *Total Token Rotation Time*.[4] The value of TTRT is given by the formula:

$$\mathrm{TTRT} = L + \sum_{i=1}^{n} \mathrm{THT}_i + \varepsilon$$

L is the propagation length of the ring and it includes repeater delays caused by all stations. The role of ε is to compensate for the variability of the repeater delays. Thus, TTRT represents a slightly inflated upper bound on the actual value of the token rotation time.

Similarly as in FDDI, each station is equipped with a timer called the *Token Rotation Timer* (TRT) which counts the time elapsed since the moment the station last acquired the token. Unlike in FDDI, TRT is not used for synchronous traffic (in fact, no notion of synchronous traffic is needed in DPMA), but as a pointer to estimate the token position. Whenever a station acquires the token, it resets this timer to 0. As TTRT is slightly pessimistic, the token arrives at a station a moment before its TRT timer equals TTRT; however, this difference is small and no phenomenon similar to the *early token* in FDDI can occur.

The rules for transmitting are as follows:

1. Stations willing to transmit constantly check for silence in their primary and secondary locations. Possession of the token is not relevant.
2. Whenever a station willing to transmit a packet senses a period of silence in the primary location, it starts transmitting, first the preamble, then the packet itself.
3. Whenever a station willing to transmit senses some activity in its primary location, but senses silence in its secondary location; if the condition $\mathrm{TRT} > \mathrm{TTRT} - L$ holds, the station may transmit in its secondary location; otherwise, it resumes checking in step 1.
4. When a transmitting station senses the arrival of another packet, sent by an upstream station (a collision), it immediately stops transmitting and resumes the checking loop in step 1.
5. Having completed a successful packet transmission, the station immediately resumes the checking loop 1.

The collision resolution protocol can be illustrated by the following example:

> Suppose that a station S_i is transmitting a packet. Before it finishes the transmission, S_i senses (using the receiver adjacent to the active transmitter) another transmission arriving from upstream, from station S_j. As the carrier is unidirectional, S_j does not know about this collision and continues transmitting. S_i stops its transmission. Note that the packet sent by S_j is not destroyed, as it is preceded by a preamble; only the first bits of this preamble interfere with the packet being sent by S_i, which causes no problems, as the preamble has more than 10 superfluous bits.

Note that the preamble of a packet coming from upstream can only be disturbed once, irrespective of how many times this preamble triggers a collision.

The rules for packet removal are the same as in FDDI:

- The station holding the token removes from the carrier all the traffic reaching it in the primary location.
- Every station is responsible for stripping the packets that it created. This stripping is performed at either

---

[4] Note that in FDDI, TTRT stands for *Target Token Rotation Time*. Our TTRT represents an interval that is always met rather than a "target" not to be exceeded.

location, upon the recognition of the station's address in the sender field.

The receiver is guaranteed to have seen the packet before it is stripped, as every station has a receiver located between the sender's two locations. Stripping leaves a short header, which proceeds down the carrier until it is absorbed by the token-holding station. This remnant of a stripped packet is followed by a period of silence which can be reused to send another packet (as stated by the rules for transmission)—steps 2 and 3 above.

Keeping track of the primary location is simple: after a station releases the token, it swaps the primary and secondary locations. Assume that there are N stations in the network. It is a protocol invariant that the first N taps downstream from the token, including the station holding the token, are primary locations. Thus, the last N taps are secondary locations. If this property holds when the protocol is started, it is naturally preserved whenever the token is passed. Consider the situation depicted in figure 4.

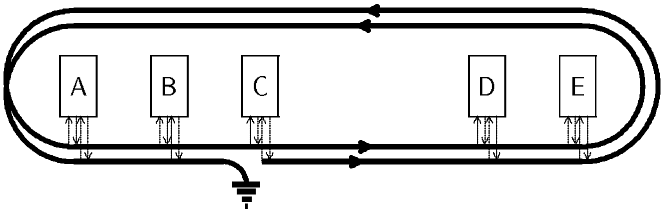

Figure 4. Station C has the token.

In this situation, the carrier may be visualised as a bus, to which each station is connected twice, as shown in figure 5:

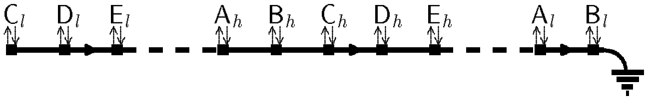

Figure 5. The PRETZEL RING flattened.

The subscripts in the station identifiers represent the carrier segments: C_l represents the connection between station C and the lower[5] LR segment. In the situation shown in the figure, C_l to B_h are primary locations, while C_h to B_l are secondary locations.

In the scenario shown in figure 5, station C will have to pass the token forward (i.e. to D) when its THT time elapses. When C releases the token, it will switch its primary location from C_l to C_h and reset its TRT. The primary locations of other stations remain unchanged. After C transmits the token, the network will look as depicted in figure 6.

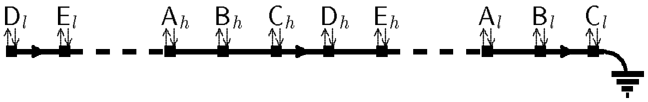

Figure 6. After the token passed to D.

A station sensing a period of silence in its primary location is allowed to transmit even if it does not possess the token. If the transmission is successful, the station "knows" that its packet will make it to the destination. Firstly, the path from the station's primary location to the secondary location is never disconnected by the token holding station. Secondly, upstream transmissions preempt downstream transfer attempts.

The station holding the token is guaranteed a successful transmission on its primary location. As the token is passed around, each station gets its chance to transmit without preemption. When the traffic is low, most "spontaneous" transmissions are successful; thus, the network avoids the token acquisition delay.

The protocol permits stations to transmit from their secondary locations, if a specific condition holds. The rationale for this condition can be explained using figure 6. Assume that at time t_0, station B wants to transmit a packet and senses silence in its secondary location B_l (but no silence in the primary location B_h). If B_l transmits its packet at this moment, three outcomes are possible:

1. A collision will occur before B finishes transmitting. B will retransmit the packet at a later time, choosing the transmission location according to the protocol.
2. The packet will reach its destination and will be received.
3. Before reaching its destination, the packet will reach the token-holding station and will be destroyed.

The third case is to be avoided. It is most likely to occur when the receiver is the upstream neighbour of B_h, i.e. A_h (note that all the other receivers of packets sent by B_l are upstream from this location). A_h will receive the packet only if the token passed through it and reached B_h beforehand. This is guaranteed to happen at time $t_0 + \text{TTRT} - t$, where t is the current reading of the TRT of station B. If a packet is sent by B_l at time t_0, it will reach B_h at time $t_0 + L$, hence the condition for success:

$$\text{TTRT} - t < L$$

As long as a station possesses the token, it doesn't use the secondary location for transmission. A station possessing the token doesn't hear any activity (other than its own transmission) in the primary location and, according to rule 3, it transmits from the primary location.

Note that the protocol requires no knowledge about the locations of stations, their number or the values of other stations' THT. Likewise, no attempt is made to guess the current location of the token.

As DPMA is a collision protocol, it may be reasonable to reduce the maximum packet length, as the longer the packet, the greater the chance that it will collide with an upstream transmission and be wasted. In the experiments reported in the next section, this maximum packet length was set to 8192 bits (1024 bytes) which is substantially less than for FDDI.

DPMA does not explicitly differentiate between synchronous and asynchronous traffic. With each turn of the

---

[5] Lower is, of course, just an abstract concept.

token, every station gets a guaranteed share of the bandwidth which it can use as it pleases. It seems a natural choice to try to transmit asynchronous traffic without the token and use the token for synchronous traffic first. Note that the jitter for synchronous traffic will be very low as each station receives the token at very regular intervals.

C. Implementing DPMA *using* FDDI *components*

DPMA requires a slightly different topology than FDDI, but one that is just as easy to construct. Likewise, the other components can be derived from FDDI components after some modifications:

- The transmission rules use a different algorithm that includes collision handling.
- The token-passing rules differ only in one aspect: no early token release or arrival.
- The definition of the TTRT and the TRT is similar, but their interpretation by the transmission algorithm is different.
- Packet removal and stripping are exactly the same as in FDDI.
- The packet format may remain identical to that of FDDI. It is slightly more efficient, though, to add one field to the header: an additional 8-bit field showing the packet length, e.g., expressed in 32-bit groups.

The only non-trivial modification is the collision detection part. It seems that the present length of the packet preamble (64 bits) is sufficient to detect upstream transmissions and yield to them without destroying too much of the preamble. Should it become a problem, the preamble length can be increased to a safe value.

The "packet length" field in the packet header could be used for tailoring efficiently the size of a packet when it is replacing a "stripped" frame; thus, packets transmitted in "stripped" silence would never collide. Adding the length field is not crucial to the performance of DPMA, although it does increase throughput. Almost the same result can be achieved by limiting the maximum length of a packet transmitted from a secondary location to, say, 1/4 of the maximum length of a packet transmitted from a primary location.

Finally, even the whole concept of packet stripping and secondary location transmissions is not a necessary component of DPMA. Without it, DPMA has the same maximum throughput as FDDI (so it is not a *capacity–1* protocol), but it achieves much lower access delays, especially for light traffic.

IV. SIMULATED ENVIRONMENT

We used simulation to compare the performance of FDDI and DPMA. The networks and their protocols were modelled in LANSF ([7]). Each point in a performance curve was produced as an average of four independent experiments, each experiment consisting in transmitting 131072 messages over the network. The results presented here have been obtained in the following environments:

- Three lengths of a ring (or loop segment) are considered: 10^4 bits, 10^5 bits, and 10^6 bits.
- There are 32 stations equally spaced along the rings.
- Traffic is uniform in the sense that the probability of every pair (*sender*, *receiver*) is the same.
- The message length is drawn from an exponential distribution with a mean of 4096 bits (512 bytes).
- Two values of TTRT were used for each network length L. These values were as $L+n\times 8642$ and $L+n\times 34568$, where n is the number of stations (32). For DPMA, they represented maximum token holding times of 8642 and 34568, respectively.
- Three variations of DPMA were considered: DPMA without packet stripping and without transmitting in secondary locations (labelled d in the figures), DPMA with packet stripping, but without transmitting in secondary locations (labelled d_p), and full DPMA with packet stripping and with transmitting in secondary locations (labelled d_s).

The results obtained for other values of the above-listed parameters were consistent with those presented in this section. The three loop lengths can be viewed as gradual technological enhancements of the network towards higher transmission rates. For example, assuming a 20 km ring, the three propagation lengths represent transmission rates of $100Mb/s$, $1Gb/s$, and $10Gb/s$.

The number of stations equal 32 seems near to a typical configuration of FDDI. A station in FDDI is usually a concentrator that connects a slower LAN (e.g., ETHERNET) to the ring; thus, the actual number of stations is much higher.

The mean message length of 4096 bits was chosen to be close to the length of a typical message transferred in a local network. Even if this length seems accidental, it has little impact on the performance measures considered, especially for a ring much longer than 4×10^3 bits (as in our case).

The token holding time of 8642 bits gives a station in DPMA enough room to transmit two average packets (4096 bits) furnished with their headers and preambles or, alternatively, one maximum length packet together with a very short packet. It seems unreasonable to use a smaller value of THT, thus 8642 bits was used as a unit of the token holding time. The two values of THT assumed in the experiments correspond to 1 unit and 4 units.

One can think of a version of DPMA whose complexity is between d and d_p. This version (which we denote by d_n) does not strip packets, but it uses secondary locations for transmission. The performance of d_n turns out to be so close to d that it does not warrant a separate consideration. Clearly, the two versions achieve the same throughput; d_n shows marginally lower access delays in the medium range of traffic conditions.

Undoubtedly, selecting a different environment would yield slightly different results, but not necessarily a different behaviour (in qualitative terms). Our claim is that the only parameter that has a significant impact on the behaviour of the network is its length (in bits).

V. PERFORMANCE MEASURES

We considered choosing the following measures of protocol performance:

- Mean packet access time, which is the mean time elapsing from the moment when station becomes ready to transmit a packet to the moment when it finishes transmitting.
- Mean message waiting time, which is the mean time elapsing between the moment a message is enqueued for transmission and the moment when its last bit is successfully transmitted.

These measures are functions of the effective throughput, as well as of the properties of the protocol (and the network's size, topology, etc.).

Logically, the unit of transmission is the message; physically, transmission is done in packets. We could measure both the mean message waiting time and the mean access time of individual packets. Normally, the two measures go hand in hand and show the same tendencies. Alas, this is not the case for FDDI, as increasing the value of TTRT artificially reduces the mean packet access time in an oversaturated network: once a station receives the token, it will transmit a large number of packets, among which only the first will have a non-zero access time. For very long networks, this phenomenon makes the mean packet access time drop as throughput increases, which demonstrates the limitations of this performance measure. In consequence, we used one measure only: the mean message waiting time as a function of throughput achieved.

In this paper, we focus on performance measures, such as maximum throughput, access delay, and network fairness. One should also remember that network protocols have additional properties, that make them different from other protocols in some other sense, e.g., FDDI and DPMA have the advantage over DQDB that a transmitting station does not have to know where the receiver station is. This property is of little relevance in static networks, but it is crucial in highly dynamic configurations.

VI. Comparison

Each of the performance measures considered here may serve as a witness to both the ability to handle congestion and the fairness of a network protocol.

A. Throughput

Let p_f denote the average packet length in FDDI and p_d denote the average packet length in DPMA (the two are not the same, as packets in DPMA are limited to 8 Kb, while packets in FDDI are limited to 36 Kb; on the other hand, the average message lengths are, of course, the same). Furthermore, let h denote the length of a packet header in FDDI, including the preamble; the packet header in DPMA (versions d_p and d_s) is 8 bits longer, as it contains an additional *packet length* field.

The maximum effective throughput of FDDI may be derived formally; it is equal to:

$$(1-\frac{L}{\mathrm{TTRT}})\times(1-\frac{h}{p_f})$$

The maximum throughput of DPMA is hard to derive for a non-slotted version.[6] For a sufficiently long network, there exists a suitable value of TTRT, such that the maximum throughput of DPMA approaches asymptotically

$$2-3\times\frac{h+8}{p_d}$$

This formula is of little consequence in practice—realistic values of parameters yielded maximum throughput values that did not exceed 1.7. Also, DPMA without stripping has precisely the same maximum throughput as FDDI.

B. Message waiting time and throughput

The relationship between effective throughput and the message waiting time for the three ring network lengths is examined in this section. Besides the network length, there is another parameter implicit to this relationship: the value of TTRT.

Figures 7 and 8 show the message delay for the *Small–a* network (10^4 bits, i.e. $a = 2.5$). The difference between FDDI and DPMA without stripping is not very pronounced. Under light load, FDDI is slightly worse; for moderate loads, it is the other way around; for heavy loads, they are similar. DPMA with packet stripping is superior under any load, even though it seldom gives stations a chance to transmit on their secondary locations.

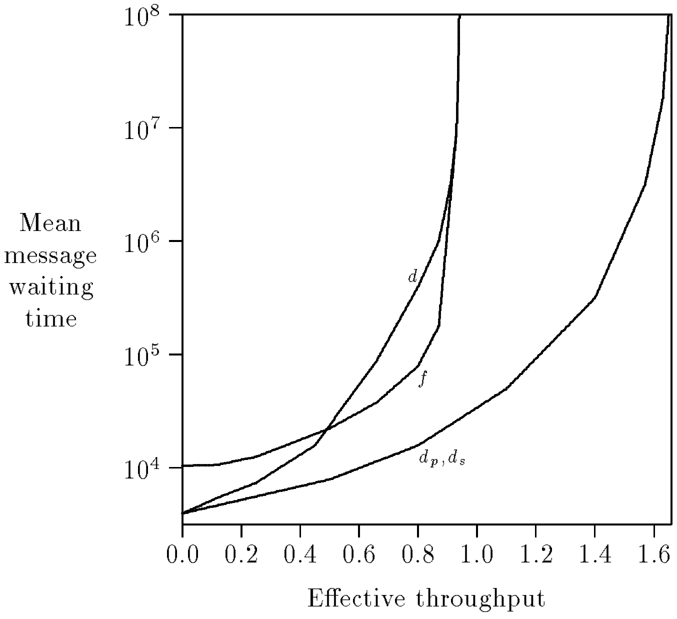

Figure 7. Delay vs. throughput (10Kb ring, TTRT=$10^4 + 32 \times 8642$ bits).

The usefulness of secondary locations for transmission is insignificant for a short ring and increases as the ring becomes longer. For each station, the fraction of the time when the secondary location is useful is equal to:

$$\frac{L}{\mathrm{TTRT}}$$

which gives 3.5% for the network from figure 7 and 0.9% for the other network. Note that irrespective of the above percentage, some stripped packet frames can be reused in

---

[6] If slots are used, the maximum throughput of DPMA is $2 - 2h/p_d$, same as DQDB.

primary locations. The performance of DPMA with packet stripping depends on the choice of TTRT to an extent much greater than FDDI.

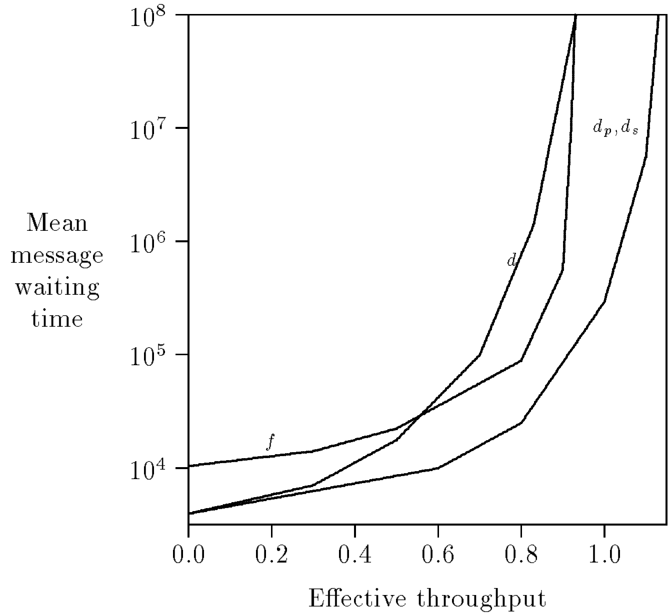

Figure 8. Delay vs. throughput (10Kb ring, TTRT=$10^4 + 32 \times 34568$ bits).

The not-so-spectacular (in comparison to FDDI) performance of d for medium range of traffic conditions is caused by two factors: the smaller maximum packet length and, more importantly, the more even and deterministic token allocation. This property tends to disappear when the ring becomes longer.

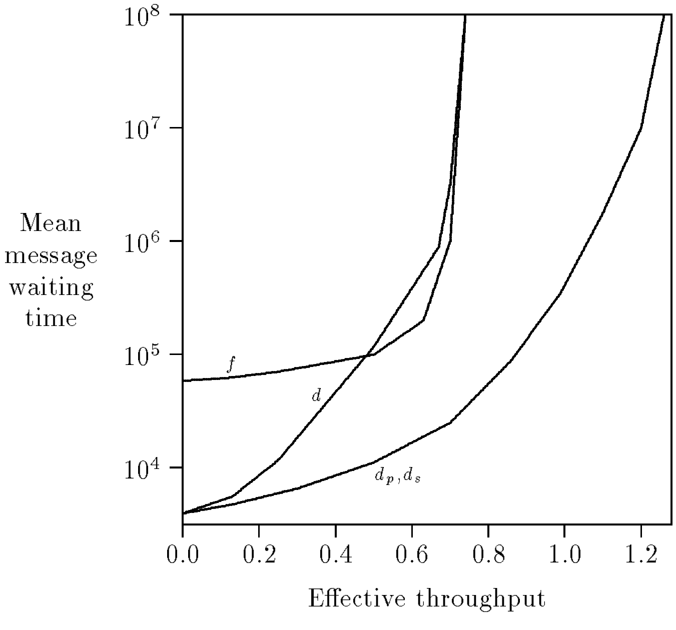

Figure 9. Delay vs. throughput (100Kb ring, TTRT=$10^5 + 32 \times 8642$ bits).

Figures 9 and 10 are for the *Medium–a* network (10^5 bits, i.e. $a = 25$). For this length, FDDI is much worse for light load; for heavy load, the relative performance of the protocols depends mainly on the choice of TTRT, although DPMA with packet stripping remains superior for every load.

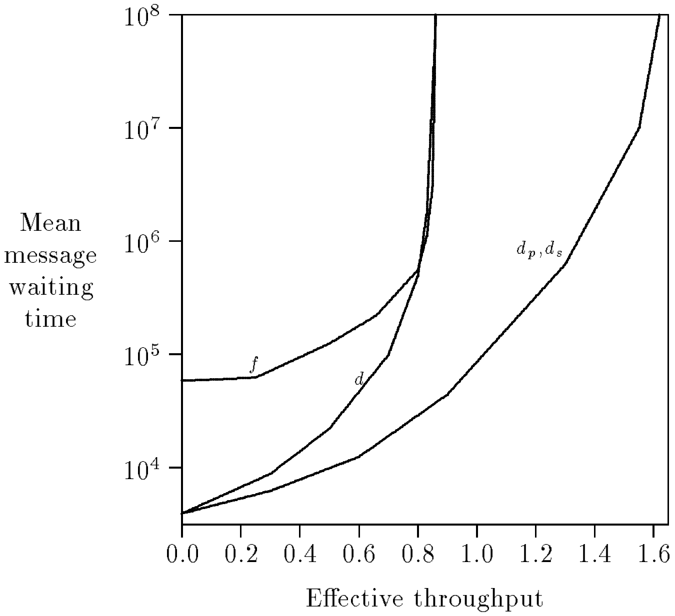

Figure 10. Delay vs. throughput (100Kb ring, TTRT=$10^5 + 32 \times 34568$ bits).

The performance of the d_p variant of DPMA for the *small* and *medium–a* networks is not much worse than the performance of d_s. The two variants achieve identical throughput and the negligible superiority of d_s for medium traffic conditions is not pronounced well enough to warrant a separate curve. This suggests that the concept of secondary location transmissions is of little relevance for these networks. The situation changes when we look at the behaviour of the *Big–a* network (figures 11 and 12).

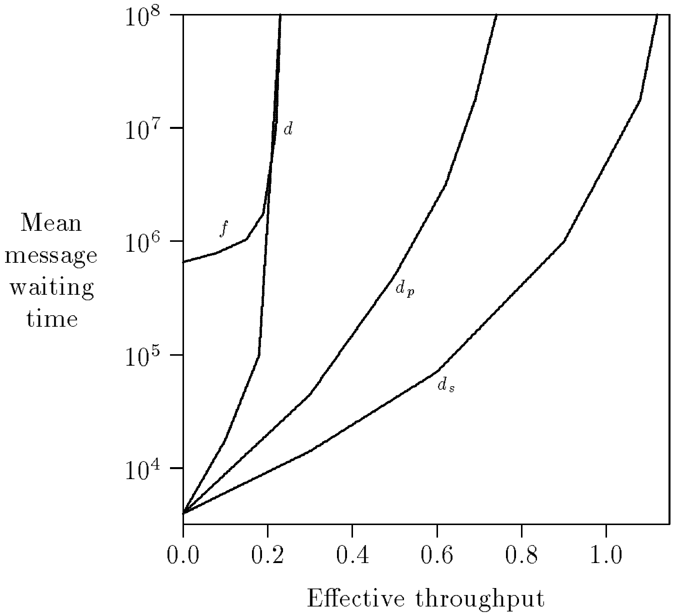

Figure 11. Delay vs. throughput (1Mb ring, TTRT=$10^6 + 32 \times 8642$ bits).

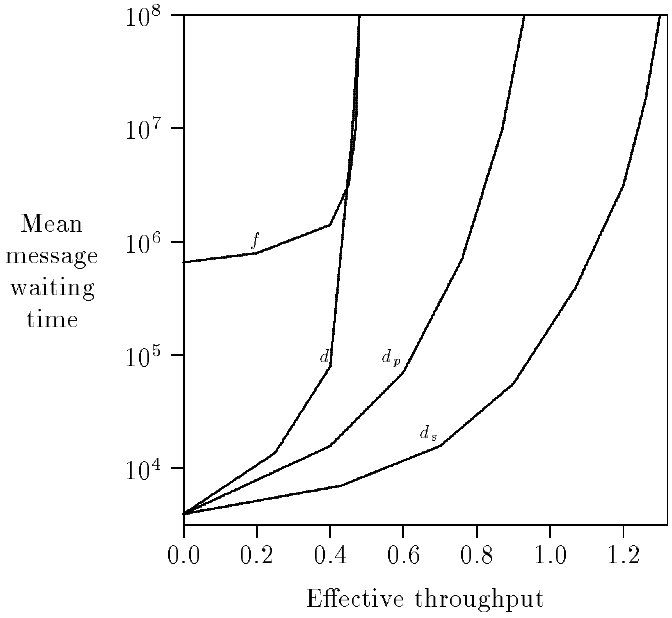

Figure 12. Delay vs. throughput (1Mb ring, TTRT=$10^6 + 32 \times 34568$ bits).

For the *Big–a* network (10^6 bits, i.e. $a = 250$), FDDI is outclassed, even by the simplest variant of DPMA. The difference between d_p and d_s becomes clearly visible and the impact of secondary location transmissions begins to count.

VII. Summary

We presented a network topology and a MAC–level protocol for a network designed to compete with FDDI. Our protocol has a better performance than FDDI, especially for high-speed (or very long) networks.

The proposed network topology may be constructed by a simple adaptation of the dual counter-rotating ring topology used in many commercial implementations of FDDI ([8]). Except for the collision sensing mechanism, the protocol requires no features that are not in use in FDDI.

This result calls for further work. Namely, a slotted version should be considered and compared with DQDB. The proposed protocol is fair and its maximum throughput is the same as that of DQDB.

References

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

[2] Distributed Queue Dual Bus Subnetwork of a Metropolitan Area Network. IEEE Std 802.6–1990, July 1991.

[3] J. Limb and C. Flores. Description of Fasnet, A unidirectional local area communications network. *Bell Systems Technical Journal*, Sept. 1982.

[4] P. Jacquet and P. Muhlender. Machnet: A Simple Access Protocol for High Speed or Long Haul Communications. In *SIGCOMM'90 Symposium*, pages 90–100, Sept. 1989.

[5] J. Limb. A Simple Multiple Access Protocol for Metropolitan Area Networks. In *SIGCOMM'90 Symposium*, pages 69–78, Sept. 1990.

[6] H. Mueller, M. Nassehi, and J. Wong. DQMA and CRMA: New access schemes for Gbit/s LANs and MANs. In *Proceedings of INFOCOM'90*, San Francisco, CA, June 1990.

[7] P. Gburzyński and P. Rudnicki. LANSF: a protocol modelling environment and its implementation. *Software Practice and Experience*, 21(1):51–76, Jan. 1991.

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